

**Министерство образования и науки Республики Татарстан
Государственное автономное профессиональное образовательное учреждение
«Сабинский аграрный колледж»**

РАССМОТРЕНО на заседании
предметной (цикловой) комиссии
Протокол № 4 от 15 апреля 2026 года
ОДОБРЕНО
на заседании Педагогического совета
ГАПОУ «САК»
протоколом №29 от «16» апреля 2026 года

**ФОНДЫ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ
ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
ПО ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ**

ПМ.02 Разработка и интеграция модулей программного обеспечения

по специальности

09.02.11 «Разработка и управление программным обеспечением»



**ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ**

СВЕДЕНИЯ О СЕРТИФИКАТЕ ЭП

Сертификат: 1E093667AA5CBF33044A6008CC579BE5
Владелец: Бикмухаметов Закиржан Миннемуллович
Действителен с 02.09.2025 до 26.11.2026

1. Результаты освоения модуля, подлежащие проверке на экзамене по модулю

Профессиональные и общие компетенции	Показатели оценки результата
ПК 2.1. Проектировать модули программного обеспечения.	– Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием. – Оформлять документацию на программные средства. – Оценивать сложность алгоритма. – Определяет основные этапы разработки программного обеспечения – Использует основные принципы технологии структурного и объектно-ориентированного программирования
ПК 2.2. Разрабатывать модули программного обеспечения.	– Использует основные принципы технологии структурного и объектно-ориентированного программирования – Создает программный код по разработанному алгоритму как отдельный модуль – Использует основные принципы технологии структурного и объектно-ориентированного программирования – Оформляет документацию на программные средства. – Осуществляет разработку кода программного модуля на языках высокого уровня, в том числе для мобильных платформ – Использует API современных мобильных операционных систем
ПК 2.3. Выполнять интеграцию модулей и компонентов программного обеспечения.	– Использует систему контроля версий. – Получает код с заданной функциональностью и степенью качества. Организует заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов. – Использует различные транспортные протоколы и стандарты форматирования сообщений. – Организует постобработку данных. – Создает классы-исключения на основе базовых классов. Выявляет ошибки в системных компонентах на основе спецификаций.

	– Использует эффективные приемы работы в системах контроля версий.
ПК 2.4. Выполнять тестирование и отладку программного обеспечения.	– Выполняет ручное и автоматизированное тестирование программного модуля. – Анализирует проектную и техническую документацию. – Использует инструментальные средства отладки программных продуктов. – Определяет источники и приемники данных. – Выполняет тестирование интеграции. – Организует постобработку данных. – Выполняет отладку, используя методы и инструменты условной компиляции. – Выявляет ошибки в системных компонентах на основе спецификаций. – Использует методы и схемы обработки исключительных ситуаций – идентифицирует сбои и ошибки при интеграции приложений – Оценивает размер минимального набора тестов. – Разрабатывает тестовые пакеты и тестовые сценарии. – Соблюдает принципы процесса разработки программного обеспечения
ПК 2.5. Осуществлять документирование программных модулей программного обеспечения.	– Оформляет результаты инспектирования в соответствии с требованиями программной документации
ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам.	– Распознает сложные проблемы в знакомых ситуациях. – Выделяет сложные составные части проблемы и описывает её причины и ресурсы, необходимые для её решения в целом. – Определяет потребность в информации и предпринимает усилия для её поиска. – Выделяет главные и альтернативные источники нужных ресурсов. – Разрабатывает детальный план действий и придерживается его. – Оценивает результат своей работы, выделяет в нём сильные и слабые стороны. – Качество результата решения ситуационной задачи, в целом, соответствует требованиям.

ОК 02. Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности.	– Планирует информационный поиск из широкого набора источников, необходимого для выполнения профессиональных задач. – Проводит анализ полученной информации, выделяет в ней главные аспекты. – Структурирует отобранную информацию в соответствии с параметрами поиска. – Интерпретирует полученную информацию в контексте профессиональной деятельности.
ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях.	– Использует актуальную нормативно-правовую документацию по специальности. – Применяет современную научно профессиональную терминологию. – Определяет траекторию профессионального развития и самообразования.
ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.	– Грамотно устно и письменно излагает свои мысли по профессиональной тематике на государственном языке. – Проявляет толерантность в рабочем коллективе.
ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения.	– Понимает значимость своей профессии (специальности). – Демонстрирует поведение на основе общечеловеческих ценностей.
ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях.	– Соблюдает правила экологической безопасности при ведении профессиональной деятельности. – Обеспечивать ресурсосбережение на рабочем месте.
ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.	– Сохраняет и укрепляет здоровье посредством использования средств физической культуры. – Поддерживает уровень физической подготовленности для успешной реализации профессиональной деятельности.

ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках.	– Применяет в профессиональной деятельности инструкции на государственном и иностранном языке. – Ведет общение на профессиональные темы. – Понимает общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые).
---	--

2. Типовое задание для экзаменуемого

ПЕРЕЧЕНЬ ВОПРОСОВ К ЭКЗАМЕНУ по междисциплинарному курсу МДК 02.01 «Разработка программных модулей»

1. Понятие ЖЦ ПО
2. Этапы ЖЦ ПО
3. Технология структурного программирования
4. Основные принципы объектно-ориентированного программирования
5. Виды сортировок
6. Оценка сложности алгоритмов сортировки.
7. Инкапсуляция
8. Наследование
9. Полиморфизм
10. Классы: основные понятия.
11. Перегрузка методов.
12. Перегрузка операций.
13. Иерархия классов
14. Интерфейсы
15. Синтаксис интерфейсов
16. Структуры
17. Регулярные выражения
18. Виртуальные методы и свойства
19. Элементы управления WPF
20. Принципы работы с БД
21. Доступ к данным из БД
22. Создание таблицы, работа с записями в БД
23. Ручное тестирование
24. Библиотеки классов (что такое и зачем)

ПЕРЕЧЕНЬ ПРИМЕРНЫХ ЗАДАЧ К ЭКЗАМЕНУ
по междисциплинарному курсу МДК 02.01 «Разработка программных модулей»
В практической экзаменационной задаче будут охвачены следующие тематики
по работе с базой данных и WPF:

- блок-схема (детальная);
- отображение данных из бд, а также их добавление;
- разработка библиотеки;
- выполнение вычислений по данным из полей и запись расчетных данных в бд;
- фильтрация и сортировка данных;
- формирование выходного документа word по примеру;
- ограничение размера поля и вводимых символов в поля;
- работа с капчей;
- авторизация и регистрация пользователя;
- штрихкод;
- работа с listview.

Приветствуется:

- комментарии кода;
- присутствие исключений в программном коде.

Примеры экзаменационного задания

Вариант 1

Разработайте алгоритм добавления нового заказа по ГОСТ 19.701-90

Решите производственную задачу

Данные о книгах хранятся в БД и загружаются в программу. **В базу данных информацию импортировать из файла Книги.xlsx.** Данные привести к 3 нормальной форме.

При двойном нажатии на книгу отображается подробная информация о выбранной книге, как на рисунке ниже. По нажатию на кнопки «Плюс» и «Минус» увеличивается количество. В поле для ввода количества можно вводить только числа. По нажатию на кнопку «Сформировать» происходит расчет покупки. Покупка должна записаться в базу данных, а также **должен сформироваться чек.** Шаблон чека находится в ресурсах к экзамену.



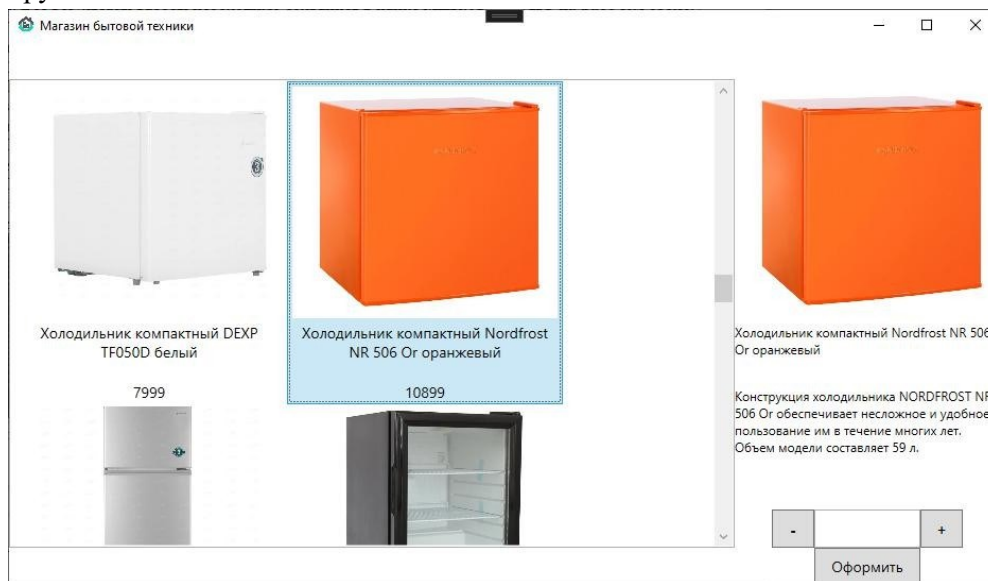
Вариант 2

Решение производственной задачи

Создать приложение средствами VisualStudio для расчета стоимости покупки.

Данные о бытовой технике хранятся в БД и загружаются в приложения в виде списка с изображением. В базу данных информацию импортировать из файла Бытовая техника.xlsx. Данные необходимо привести к 3 нормальной форме.

При двойному нажатию на выбранный элемент отображается подробная информация о бытовой технике, как на рисунке ниже. По нажатию на кнопки «Плюс» и «Минус» увеличивается количество. В поле для ввода количества можно вводить только числа. По нажатию на кнопку «Оформить» происходит расчет стоимости покупки. И формируется чек в Word.



Вариант 3

Решение производственной задачи

Идентификатор государственного контракта (ИГК) — это уникальный цифровой код, состоящий из букв и цифр. Он содержит зашифрованную информацию о документе. Добавляется к номеру контракта. Номер контракта имеет следующую структуру: ИГК/номер контракта. Пример: 1820187301181432209200424/15-04_19

Создать приложение средствами VisualStudio для проверки ИГК и номера договора, и формирования договора.

Создать библиотеку для проверки ИГК с номером договора для договоров государственного оборонного заказа. Название библиотеки IGK_LIB.dll.

1 метод boolean check_IGK(String mail) должен проверить длину строки от 20 до 25 цифр, до слеша (/) и состоящую только из цифр. После символа (/) должен быть номер договора, который может содержать цифры, латинские алфавита, спецсимволы (-, _).

2 метод проверяет boolean GET_IGK(String mail) должен проверять наличие данного номера, если такой существует добавить символ.

Данные о существующих номерах должно храниться в БД.

Подключить библиотеку к приложению для проверки работоспособности. Реализовать добавление номеров в БД и формирование договора в Word из шаблона. В документе необходимо отобразить ИГК с номером договора и сегодняшней даты со временем.

Разработать модульные тесты на каждый метод в трех условиях (нормальные, экстремальные, исключительные условия).

Экзаменационный тест по междисциплинарному курсу МДК 02.01 «Разработка программных модулей»

Время на выполнение: 45 мин.

1. Что означает аббревиатура ЖЦ ПО?

- a) Жесткий цикл программного обеспечения
- b) Жизненный цикл программного обеспечения
- c) Живой цикл программирования
- d) Журнал циклов программы

2. Какой этап НЕ входит в классический ЖЦ ПО?

- a) Анализ требований
- b) Проектирование
- c) Тестирование
- d) Продажа программы

3. Какой процесс происходит после тестирования в ЖЦ ПО?

- a) Внедрение
- b) Сбор требований
- c) Кодирование
- d) Проектирование

4. Какой принцип НЕ относится к структурному программированию?

- a) Последовательность
- b) Ветвление
- c) Цикличность
- d) Наследование

5. Какой оператор используется для ветвления в структурном программировании?

- a) if-else
- b) for

c) while

d) return

6. Какой подход запрещает использование goto?

a) Объектно-ориентированное программирование

b) Структурное программирование

c) Функциональное программирование

d) Логическое программирование

7. Какой принцип ООП означает скрытие внутренней реализации?

a) Инкапсуляция

b) Наследование

c) Полиморфизм

d) Абстракция

8. Как называется механизм, позволяющий классу-потомку использовать методы класса-родителя?

a) Инкапсуляция

b) Наследование

c) Полиморфизм

d) Перегрузка

9. Что позволяет полиморфизм?

a) Использовать один интерфейс для разных реализаций

b) Скрывать данные

c) Наследовать методы

d) Перегружать операторы

10. Что из перечисленного НЕ является частью класса?

a) Поля

b) Методы

c) Пространство имен

d) Свойства

11. Что такое перегрузка методов?

- a) Определение нескольких методов с одним именем, но разными параметрами
- b) Переопределение метода в дочернем классе
- c) Увеличение производительности метода
- d) Использование одного метода для разных классов

12. Какая операция НЕ может быть перегружена в C++?

- a) +
- b) -
- c) :: (оператор области видимости)
- d) ==

13. Что означает иерархия классов?

- a) Вложенность классов друг в друга
- b) Наследование классов от одного базового
- c) Группировка классов по пространствам имен
- d) Использование интерфейсов

14. Что из перечисленного НЕ является интерфейсом?

- a) Абстрактный класс с чистыми виртуальными методами
- b) Класс с реализованными методами
- c) interface в C#
- d) Набор методов без реализации

15. Как объявляется интерфейс в C#?

- a) class InterfaceName { ... }
- b) interface InterfaceName { ... }
- c) struct InterfaceName { ... }

d) `abstract InterfaceName { ... }`

16. Что из перечисленного НЕ является структурой?

a) `struct` в C

b) `record` в C#

c) `class` в Java

d) `tuple` в Python

17. Какое регулярное выражение соответствует числу?

a) `\d+`

b) `\w+`

c) `[a-z]+`

d) `\s+`

18. Какой алгоритм сортировки имеет сложность $O(n \log n)$ в среднем случае?

a) Сортировка пузырьком

b) Быстрая сортировка (QuickSort)

c) Сортировка выбором

d) Сортировка вставками

19. Какой язык используется для запросов к БД?

a) C#

b) SQL

c) HTML

d) XML

20. Какой SQL-запрос создает таблицу?

a) `INSERT INTO table_name ...`

b) `CREATE TABLE table_name ...`

c) `UPDATE table_name ...`

d) DELETE FROM table_name ...

21. Какой оператор SQL используется для выборки данных?

a) SELECT

b) UPDATE

c) DELETE

d) INSERT

22. Что такое ручное тестирование?

a) Тестирование без автоматизации

b) Написание unit-тестов

c) Использование Selenium

d) Проверка кода компилятором

23. Для чего нужны библиотеки классов?

a) Для хранения исполняемого кода

b) Для повторного использования кода

c) Для хранения данных пользователя

d) Для визуального оформления программы

24. Какой элемент управления WPF используется для ввода текста?

a) TextBox

b) Button

c) ListBox

d) Label

25. Что такое виртуальный метод?

a) Метод, который можно переопределить в дочернем классе

b) Метод, который нельзя изменить

c) Статический метод

d) Метод без реализации

26. Какой принцип ООП позволяет использовать один метод для разных типов данных?

a) Инкапсуляция

b) Наследование

c) Полиморфизм

d) Абстракция

27. Какой алгоритм сортировки имеет сложность $O(n^2)$ в худшем случае?

a) Быстрая сортировка

b) Сортировка слиянием

c) Сортировка пузырьком

d) Пирамидальная сортировка

28. Какой оператор SQL изменяет данные в таблице?

a) SELECT

b) UPDATE

c) DELETE

d) INSERT

29. Какой элемент WPF используется для отображения списка?

a) TextBox

b) Button

c) ListBox

d) Label

30. Что означает оценка сложности алгоритма $O(1)$?

a) Константное время

b) Линейное время

c) Квадратичное время

d) Логарифмическое время

Ключ к тесту:

1) В	11) А	21) А
2) D	12) С	22) А
3) А	13) В	23) В
4) D	14) В	24) А
5) А	15) В	25) А
6) В	16) С	26) С
7) А	17) А	27) С
8) В	18) В	28) В
9) А	19) В	29) С
10) С	20) В	30) А

Критерии оценки

Процент результативности (правильных ответов)	Оценка уровня подготовки	
	балл (отметка)	вербальный аналог
27 - 30	5	отлично
22 - 26	4	хорошо
16 - 21	3	удовлетворительно
менее 16	2	неудовлетворительно

ПЕРЕЧЕНЬ ВОПРОСОВ К ЭКЗАМЕНУ по междисциплинарному курсу МДК 02.02 «Поддержка и тестирование программных модулей»

1. Основные понятия в области тестирования программного обеспечения. Цели и задачи тестирования программного обеспечения.
2. Виды ошибок. Методы отладки. Основные понятия и классификация видов ошибок. Классификация методов отладки.
3. Методы тестирования. Тестирование «белым ящиком». Методы тестирования. Тестирование «черным ящиком». Методы тестирования. Недостатки тестирования «белым и черным ящиком».
4. Классификация тестирования по уровням. Классификация по запуску кода на исполнение.
5. Классификация тестирования по уровням. Классификация по степени важности тестируемых функций.
6. Тестирование производительности. Этапы проведения тестирования производительности. Анализ системы и подбор требований. Виды проверок на этапе проведения тестирования производительности.
7. Регрессионное тестирование. Цели и задачи регрессионного тестирования. Разбор основных методологий управления проектами. Стадии регрессионного тестирования. Автоматизация регрессионного тестирования.
8. Тестирование «белым ящиком». Понятие тестирования базового пути. Постановка задачи.
9. Тестирование «черным ящиком». Понятие разбиения по эквивалентности и анализа граничных значений.
10. Тестирование «черным ящиком». Построение дерева разбиений. Подготовка и выполнение тестовых сценариев.
11. Модульное тестирование.
12. Интеграционное тестирование.

Экзаменационный тест по междисциплинарному курсу МДК 02.02 «Поддержка и тестирование программных модулей»

Время на выполнение: 45 мин.

1. Что такое тестирование программного обеспечения?

- a) Процесс изучения кода программы
- b) Процесс выполнения программы с целью обнаружения ошибок
- c) Процесс написания документации
- d) Процесс оптимизации производительности

2. Какая основная цель тестирования ПО?

- a) Ускорение работы программы
- b) Обнаружение дефектов и повышение качества
- c) Сокращение времени разработки
- d) Упрощение интерфейса

3. Что относится к задачам тестирования?

- a) Только проверка корректности кода
- b) Проверка соответствия требованиям, поиск ошибок, оценка качества
- c) Только автоматизация тестов
- d) Только проверка UI

4. Какие виды ошибок существуют в ПО?

- a) Синтаксические, логические, runtime
- b) Только синтаксические
- c) Только runtime
- d) Только ошибки интерфейса

5. Что такое отладка (debugging)?

- a) Процесс написания кода
- b) Процесс поиска и исправления ошибок
- c) Процесс тестирования UI
- d) Процесс документирования кода

6. Какой метод отладки предполагает пошаговое выполнение программы?

- a) Логирование
- b) Трассировка
- c) Code review
- d) Статический анализ

7. Что означает тестирование «белым ящиком»?

- a) Тестирование без знания внутренней структуры
- b) Тестирование с доступом к исходному коду
- c) Тестирование только UI
- d) Тестирование производительности

8. Какой метод относится к тестированию «черным ящиком»?

- a) Анализ покрытия кода
- b) Разбиение на классы эквивалентности
- c) Проверка циклов в коде
- d) Анализ алгоритмов

9. Недостаток тестирования «белым ящиком»:

- a) Не проверяет логику программы
- b) Требует знания кода и сложен для больших систем
- c) Не выявляет ошибки интерфейса
- d) Не подходит для модульного тестирования

10. Какие уровни тестирования существуют?

- a) Модульное, интеграционное, системное, приемочное
- b) Только модульное и системное
- c) Только UI-тестирование
- d) Только нагрузочное тестирование

11. Что проверяет модульное тестирование?

- a) Взаимодействие модулей
- b) Отдельные функции или методы
- c) Только производительность
- d) Только безопасность

12. Что включает тестирование производительности?

- a) Только скорость загрузки страницы
- b) Нагрузочное, стрессовое, стабильности
- c) Только проверку UI
- d) Только анализ кода

13. Какой этап тестирования производительности идет первым?

- a) Нагрузка системы
- b) Анализ требований
- c) Автоматизация тестов
- d) Исправление ошибок

14. Цель регрессионного тестирования:

- a) Проверить новые функции
- b) Убедиться, что изменения не сломали старый функционал
- c) Только оптимизировать код
- d) Только тестировать базы данных

15. Когда выполняется регрессионное тестирование?

- a) Только в начале проекта
- b) После внесения изменений в код
- c) Только перед релизом

d) Только при нагрузочном тестировании

16. Что такое тестирование базового пути?

- a) Проверка всех возможных путей выполнения кода
- b) Тестирование только главной функции
- c) Проверка UI
- d) Тестирование API

17. Какой метод используется для определения сложности кода в тестировании базового пути?

- a) Анализ граничных значений
- b) Цикломатическая сложность
- c) Таблица решений
- d) Диаграмма состояний

18. Что такое классы эквивалентности?

- a) Группы входных данных, обрабатываемых одинаково
- b) Разные версии программы
- c) Типы ошибок в коде
- d) Методы тестирования UI

19. Какой пример граничного значения для поля "возраст" (допустимый диапазон: 18–65)?

- a) 18, 65
- b) 0, 100
- c) 10, 70
- d) 20, 60

20. Для чего используется дерево разбиений?

- a) Для визуализации структуры кода

- b) Для систематического создания тестовых случаев
- c) Для анализа производительности
- d) Для документирования требований

21. Какой элемент НЕ входит в подготовку тестовых сценариев?

- a) Определение входных данных
- b) Описание ожидаемого результата
- c) Проверка скорости работы программы
- d) Указание шагов выполнения

22. Какой инструмент используется для модульного тестирования в Python?

- a) Selenium
- b) JUnit
- c) pytest
- d) Postman

23. Что проверяет модульный тест?

- a) Взаимодействие нескольких модулей
- b) Корректность работы одной функции или метода
- c) Производительность системы
- d) Безопасность API

24. Что является целью интеграционного тестирования?

- a) Проверка отдельного модуля
- b) Проверка взаимодействия модулей
- c) Тестирование UI
- d) Проверка нагрузки на сервер

25. Какой подход используется в интеграционном тестировании?

- a) Нисходящий (Top-Down)

- b) Восходящий (Bottom-Up)
- c) Сэндвич (Sandwich)
- d) Все перечисленные

26. Что такое smoke-тестирование?

- a) Проверка основных функций после сборки
- b) Тестирование безопасности
- c) Нагрузочное тестирование
- d) Тестирование документации

27. Какой вид тестирования проверяет устойчивость системы при высокой нагрузке?

- a) Регрессионное
- b) Стрессовое
- c) Модульное
- d) Интеграционное

28. Что проверяет приемочное тестирование?

- a) Соответствие требованиям заказчика
- b) Скорость работы алгоритмов
- c) Корректность отдельных функций
- d) Взаимодействие модулей

29. Какой метод НЕ относится к тестированию черным ящиком?

- a) Анализ граничных значений
- b) Таблица решений
- c) Покрытие условий (condition coverage)
- d) Причина-следствие (cause-effect)

30. Что такое «ложноположительный» результат в тестировании?

- a) Тест проходит, хотя должен был упасть

- b) Тест падает, хотя должен был пройти
- c) Тест не запускается
- d) Тест выполняется слишком долго

Ключ к тесту:

1) В	12) В	23) В
2) В	13) В	24) В
3) В	14) В	25) D
4) А	15) В	26) А
5) В	16) А	27) В
6) В	17) В	28) В
7) В	18) А	29) А
8) В	19) А	30) С
9) В	20) В	31) А
10) А	21) С	
11) В	22) С	

Критерии оценки

Процент результативности (правильных ответов)	Оценка уровня подготовки	
	балл (отметка)	вербальный аналог
27 - 30	5	отлично
22 - 26	4	хорошо
16 - 21	3	удовлетворительно
менее 16	2	неудовлетворительно

Пример экзаменационного задания по МДК 02.02 «Поддержка и тестирование программных модулей»

- Проведите тестирование представленной вам программы методом «черного ящика».
- Разработайте 3 тест-кейса на каждый тип условия: в нормальных, исключительных и экстремальных; используя шаблон для тест-кейсов на сервере. Суммарно у вас должно получиться 9 тест-кейсов.
Приложение и тест-кейсы для тестирования: \\server\VsemInfo\Учебная работа\09.02.11 Разработка и управление программным обеспечением\Экзамен ПМ.02
-

ПЕРЕЧЕНЬ ВОПРОСОВ ДЛЯ ПОДГОТОВКИ К ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

По междисциплинарному курсу МДК 02.03 «Технология разработки программного обеспечения»

1. Понятия требований, классификация, уровни требований. Методологии и стандарты, регламентирующие работу с требованиями.
2. Современные принципы и методы разработки программных приложений. Методы организации работы в команде разработчиков. Системы контроля версий
3. Основные подходы к интегрированию программных модулей. Стандарты кодирования.
4. Описание требований: унифицированный язык моделирования - краткий словарь. Диаграммы UML.
5. Описание и оформление требований (спецификация). Анализ требований и стратегии выбора решения
6. Диаграмма Вариантов использования.
7. Диаграмма Последовательности.
8. Диаграмма Деятельности.
9. Диаграмма Кооперации.
10. Диаграмма Развертывания.
11. Диаграмма Состояний.
12. Диаграмма Классов.
13. Диаграмма Объектов.
14. Диаграмма Поточков данных.
15. Диаграмма Компонентов.
16. Цели и задачи и виды тестирования. Стандарты качества программной документации. Меры и метрики.
17. Тестовое покрытие. Тестовый сценарий, тестовый пакет.
18. Анализ спецификаций. Верификация и аттестация программного обеспечения.

Зачетное задание по МДК.02.03 Технология разработки программного обеспечения

Вариант 1

1. Укажите название модели разработки программного обеспечения

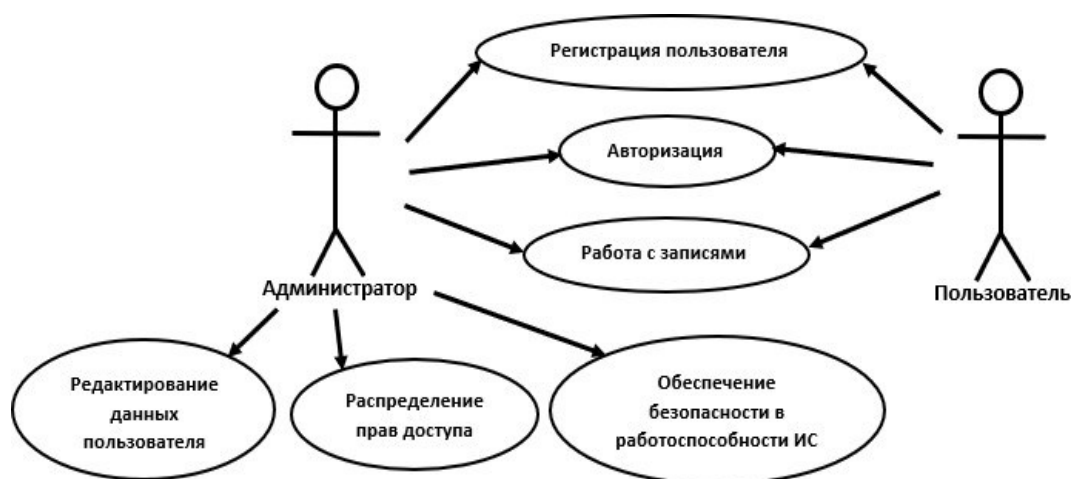


Какие плюсы и минусы можно выделить для данной модели?

2. Укажите название диаграммы. Опишите ошибки в оформлении данной диаграммы.



3. Напишите, как называется данная диаграмма? Перечислите ошибки в оформлении.



4. Проведите анализ предметной области и разработайте ER-диаграмму:

Требуется разработать информационную систему для компании, которая занимается издательской деятельностью. ИС должна содержать данные о сотрудниках компании, книгах, авторах, финансовом состоянии компании и предоставлять возможность получать разнообразные отчёты.

В системе должны быть реализованы следующие функции:

- просмотр списка книг и полной информации о книге, находящихся в печати;
- просмотр списка редакторов, работающих над книгами;
- оформление продажи книги.

5. Для предметной области из задания 4 разработать диаграмму последовательности

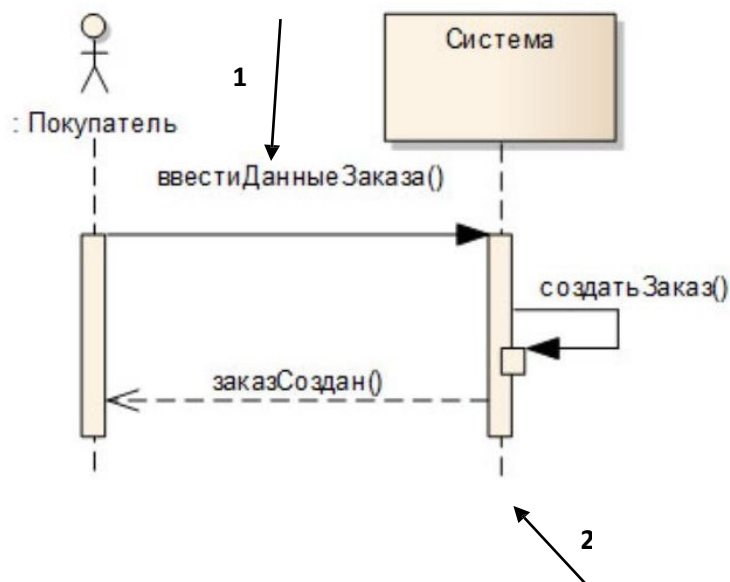
6. Какие виды тестирования вы знаете? К какому виду тестирования относится нагрузочное тестирование?

7. Опишите метод тестирования программного продукта «Белый ящик»

8. Выберите задачи, которые входят в процесс сопровождения программного продукта

- a. Нагрузочное тестирование
- b. Адаптация ПО для возможности работы на другой аппаратной платформе
- c. Расширение функциональных возможностей
- d. Улучшение дизайна
- e. Устранение сбоев

9. Напишите, как называется диаграмма. Укажите названия основных элементов



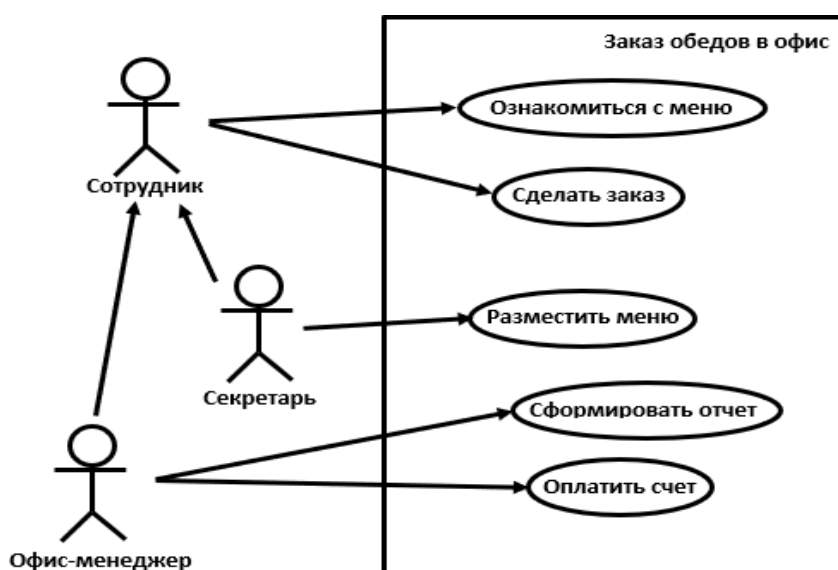
Вариант 2

1. Как называется данная модель разработки программного обеспечения?

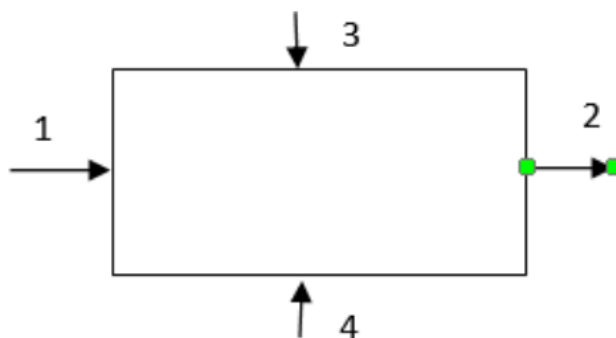


Какие плюсы и минусы можно выделить для данной модели?

2. Напишите, как называется данная диаграмма? Перечислите указанные на диаграмме прецеденты



3. Для функциональной диаграммы указать названия потоков (в общем виде)



4. Проведите анализ предметной области и разработайте ER-диаграмму:

Требуется разработать информационную систему для организации, которая предоставляет услуги в области проката транспортных средств. Организация имеет большой парк автотранспортных средств. Клиентами организации могут быть частные лица, имеющие водительское удостоверение и паспорт, удостоверяющий личность.

В договоре указываются все условия аренды (начало аренды, срок, контактное лицо организации для связи при возникновении каких-либо нештатных ситуаций и др.), условия получения и возврата автотранспортного средства, могут быть указаны различные дополнительные условия. Договор считается вступившим в силу только после его оплаты.

В системе должны быть реализованы следующие функции:

- просмотр списка автомобилей и полной информации о транспортном средстве;
- просмотр списка клиентов компании;
- оформление аренды автомобиля;
- оформление возврата автомобиля.

5. Для предметной области из задания 4 разработать диаграмму последовательности

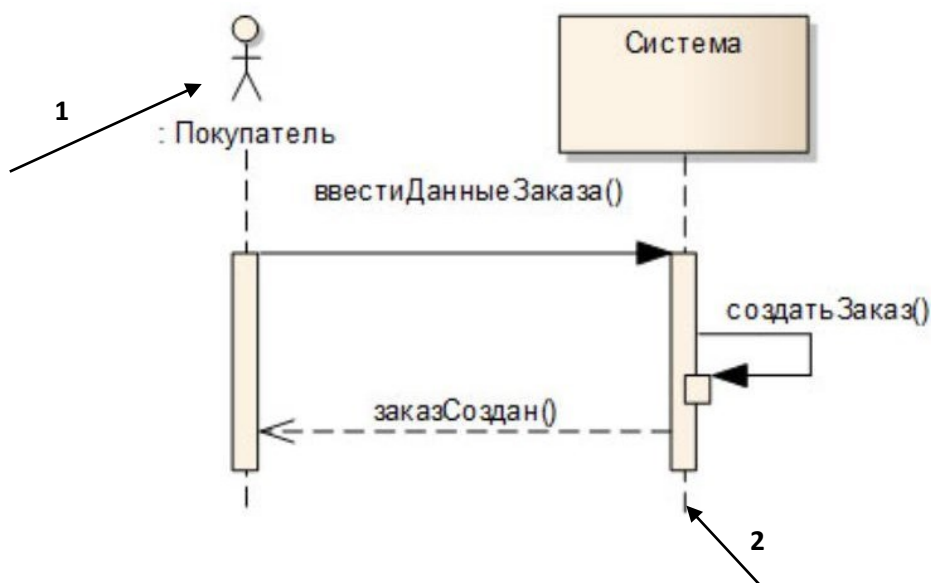
6. Какие виды тестирования вы знаете? К какому виду тестирования относится регрессионное тестирование?

7. Опишите метод тестирования программного продукта «Черный ящик»

8. Выберите задачи, которые входят в процесс сопровождения программного продукта

- а. Нагрузочное тестирование
- б. Адаптация ПО для возможности работы на другой аппаратной платформе
- в. Расширение функциональных возможностей
- г. Улучшение дизайна
- д. Устранение сбоев

9. Напишите, как называется диаграмма. Укажите названия основных элементов



Зачетное задание по МДК.02.03 Технология разработки программного обеспечения

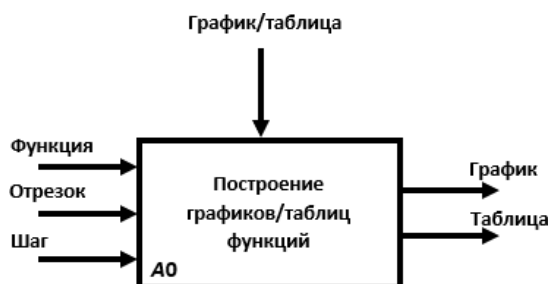
Вариант 3

1. Укажите название модели разработки программного обеспечения



Какие плюсы и минусы можно выделить для этой модели?

2. Напишите, как называется данная диаграмма. Определите, какого звена (потока) не хватает на этой диаграмме?

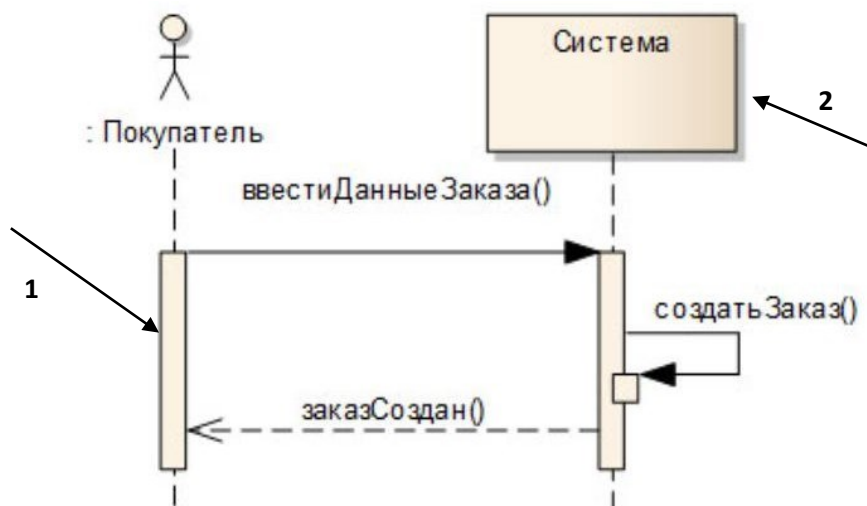


3. Укажите название диаграммы. Опишите ошибки в оформлении данной диаграммы, подпишите отношения, обозначенные цифрами



4. Проведите анализ предметной области и разработайте ER-диаграмму:
Требуется разработать информационную систему для Отдела кадров компании. ИС должна содержать данные о сотрудниках компании, подразделениях, в которых они работают.
В системе должны быть реализованы следующие функции:

- просмотр списка сотрудников и полной информации о каждом сотруднике организации;
 - прием сотрудников на работу;
 - увольнение сотрудников;
 - формирование списка сотрудников, которые в этом году выходят на пенсию.
5. Для предметной области из задания 4 разработать диаграмму последовательности
 6. Какие виды тестирования вы знаете? К какому виду тестирования относится тестирование взаимодействия?
 7. Опишите метод тестирования программного продукта «Белый ящик»
 8. Выберите задачи, которые входят в процесс сопровождения программного продукта
 - a. Нагрузочное тестирование
 - b. Адаптация ПО для возможности работы на другой аппаратной платформе
 - c. Расширение функциональных возможностей
 - d. Улучшение дизайна
 - e. Устранение сбоев
 9. Напишите, как называется диаграмма. Укажите названия основных элементов



Ответы

Вариант 1

1. Каскадная модель
 - a. Плюсы: четкая документация процесса; точное определение бюджета; определение сроков сдачи проекта; низкая степень зависимости от человеческого фактора
 - b. Минусы: длительные сроки от старта проекта до предоставления первого результата; большой объем документов; длительные согласования требований и промежуточных документов; невозможность внесения изменений в динамическом режиме.
2. Функциональная диаграмма
 - a. Ошибки: правила приема – управление, отчеты по приему – выходные данные, приемная комиссия – механизм
3. Диаграмма прецедентов
 - a. Ошибки: нет границ и названия системы
4. Издательство
 - a. Объекты: сотрудники (номер, фео, должность), книга (номер, название, автор, статус, цена), автор (номер, фео, дата рождения)
5. Диаграмма последовательности
6. Виды тестирования: функциональное, нефункциональное, связанное с изменениями
 - a. Нагрузочное – нефункциональное
7. Белый ящик - структурное тестирование — тестирование, которое учитывает внутренние механизмы системы или компонента
8. Сопровождение: адаптация, расширение функциональных возможностей, устранение сбоев, улучшение дизайна
9. Диаграмма последовательности
 - a. 1 – сообщение, 2- временная шкала (линия жизни)

Вариант 2

1. V-образная модель
 - a. Плюсы: планирование на ранних стадиях разработки системы ее тестирования (Стадия тестирования проводится одновременно с соответствующей стадией разработки); простота в использовании
 - b. Минусы: сложность поддержки параллельных событий, невозможность внесения динамических изменений в требования на разных этапах ЖЦ,
2. Диаграмма прецедентов
 - a. Прецеденты: ознакомиться с меню, сделать заказ, разместить меню, сформировать отчет, оплатить счет
3. Функциональная диаграмма. 1 – входные данные, 2 – выходные данные, 3 – управление, 4 – механизм
4. Прокат транспортных средств
 - a. Объекты: автомобиль (номер, марка, цвет, пробег), клиент (номер, фео, водУдостоверение, паспорт (серия, номер, дата выдачи), договор (дата, номер, клиент, автомобиль, срок аренды, стоимость аренды, контактное лицо, допУсловия), возврат автомобиля (дата, номер, авто)
5. Диаграмма последовательности
6. Виды тестирования: функциональное, нефункциональное, связанное с изменениями
 - a. регрессионное – связанное с изменениями
7. черный ящик - это функциональное и нефункциональное тестирование без доступа к внутренней структуре компонентов системы.
8. Сопровождение: адаптация, расширение функциональных возможностей, устранение сбоев, улучшение дизайна

9. Диаграмма последовательности
 - a. 1 – актер, 2 – временная шкала (линия жизни)

Вариант 3

1. Инкрементная модель (эволюционная)
 - a. Плюсы: получение функционального продукта после реализации каждого инкремента, предотвращение формирования громоздких перечней требований; упрощение тестирования инкрементов
 - b. Минусы: необходимость полного функционального определения системы в начале жизненного цикла, недостаточно чёткое определение требований; отсутствие снижения общих затрат на выполнение проекта; необходимость хорошего планирования и проектирования, грамотного распределения работы;
 - c. Используется: Когда основные требования к системе четко определены и понятны. В то же время некоторые детали могут дорабатываться с течением времени. Требуется ранний вывод продукта на рынок. У продукта несколько целей.
2. Функциональная диаграмма
 - a. Не хватает механизма
3. Диаграмма прецедентов
 - a. 1 - , 2 - , 3 - , 4-
4. Отдел кадров
 - a. Объекты: сотрудники (номер, фео, дата рождения), подразделения (номер, название), должность (номер, название), приказ о приеме (номер, дата, фео, должность, подразделение), увольнение (номер, дата, фео)
5. Диаграмма последовательности
6. Виды тестирования: функциональное, нефункциональное, связанное с изменениями
 - a. Тестирование взаимодействия – функциональное
7. Белый ящик - структурное тестирование — тестирование, которое учитывает внутренние механизмы системы или компонента
8. Сопровождение: адаптация, расширение функциональных возможностей, устранение сбоев, улучшение дизайна
9. Диаграмма последовательности
 - a. 1 деятельность объекта (исполнение функции), 2 – объект системы

ПРИМЕРНОЕ ЗАДАНИЕ К ЭКЗАМЕНУ ПО МОДУЛЮ ПМ.02 РАЗРАБОТКА И ИНТЕГРАЦИЯ МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

государственное бюджетное профессиональное образовательное учреждение
«Пермский химико–технологический техникум»

Утверждаю:

Зам. директора по УМР
_____ О.В.Князева

202_ -202_ уч. г.

Экзаменационное задание

к экзамену по модулю ПМ.02

«Разработка и интеграция модулей программного обеспечения»

ИНСТРУКЦИЯ

1. Внимательно прочитайте задание.
2. Вы можете воспользоваться материалом, находящимся на сетевом ресурсе \\server\VsemInfo\Учебная работа\09.02.11 Разработка и управление программным обеспечением\Экзамен ПМ.02.
3. Все результаты работы необходимо сохранить на локальном сервере GOGS (<http://10.11.13.14:3000/>) под учетной записью в соответствии с номером ПК вашего рабочего места (логин и пароль для входа на GOGS будут выданы). Репозиторий должен иметь имя ПМ04_ФИО.

Задание 1. СОЗДАНИЕ БАЗЫ ДАННЫХ

Для дальнейшего выполнения задания вам необходимо воспользоваться скриптом, предварительно загрузив его из ресурсов. В процессе разработки приложения Вы можете изменять базу данных на свое усмотрение.

Файл скрипт хранится: \\server\VsemInfo\Учебная работа\09.02.11 Разработка и управление программным обеспечением \Экзамен ПМ.02

Задание 2. РАЗРАБОТКА ПРИЛОЖЕНИЯ

Вам предстоит доработать информационную систему, предназначенную для создания и хранения рецептов.

Создайте инфраструктуру для авторизации зарегистрированных пользователей.

Файлы макетов форм хранятся: \\server\VsemInfo\Учебная работа\09.02.11 Разработка и управление программным обеспечением \Экзамен ПМ.02

В системе будет три различных типа пользователей: Пользователь, Администратор, Шеф-повар.

Пользователи будут входить в систему, используя следующий механизм. У каждого пользователя будет свой логин и пароль. Неавторизованный пользователь в систему попадать не должен.

После успешного входа пользователя, в зависимости от его роли, происходит перенаправление в следующие подсистемы:

- Пользователь: "Интерфейс пользователя"

Заказчик может просматривать список ингредиентов и рецептов. Может просматривать рецепты.

- Шеф-повар: "Интерфейс шеф-повара"

Шеф-повару можно будет создавать рецепты.

- Администратор: "Интерфейс администратора"

Администратору предоставляется полный доступ работы с системой. Необходимо предусмотреть возможность регистрации пользователей.

**Шаблоны системы предоставлены: \\server\VsemInfo\Учебная работа\09.02.11
Разработка и управление программным обеспечением \Экзамен ПМ.02**

Задание 3. ЛИЧНЫЙ КАБИНЕТ ПОЛЬЗОВАТЕЛЯ

Разработайте личный кабинет пользователя, в котором дана возможность корректировки фамилии, пароля и фото пользователя. При загрузке фото для личного кабинета, пользователю будут доступны файлы формата .jpg, .png. Изменение роли в личном кабинете не доступно.

Система должна осуществлять проверку введенного пароля безопасность и надежность.

Пароль должен состоять из 8-20 символов. В пароле можно использовать:

- латинские буквы: a-z, A-Z;
- цифры: 0-9;
- специальные символы: ! @ # \$ % ^ & * () - _ + = ; : , . / ? \ | ` ~ [] { } ^ ' " < > ~ .

Пароль не может:

- состоять только из одних цифр;
- содержать пробелы;
- состоять из простых слов: password, love, family.
- содержать личные данные: имя, фамилию.

Реализовать механизм, при котором при сохранении измененных данных в систему у пользователя будет запрошено решение примера: $(x+y)*z$, где x, y, z числа от 0 до 20, которые задаются рандомно. Количество допустимых попыток неправильного ввода результата равно 3, далее система блокируется и возможности ввода результата не предлагает.

Лист согласования			Тип согласования: последовательное	
N°	ФИО	Срок согласования	Результат согласования	Замечания
1	Бикмухаметов З.М.		 Подписано 09.06.2026 - 09:04	-