**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**МИНИСТЕРСТВО НАУКИ, ВЫСШЕГО ОБРАЗОВАНИЯ И ИННОВАЦИЙ КЫРГЫЗСКОЙ РЕСПУБЛИКИ**

**МОО ВО Кыргызско-Российский Славянский университет имени первого Президента Российской Федерации Б.Н. Ельцина**

**Высшая школа Цифровых технологий**

# Фонд оценочных средств

по дисциплине «**Процессы управления качеством программного обеспечения**»

Уровень высшего образования

МАГИСТРАТУРА

Направление подготовки

*09.04.04 - РФ, 710400 - КР Программная инженерия*

(код и наименование направления подготовки)

*"Основы анализа и разработки приложений с большими объемами распределенных данных"* (профиля) образовательной программы)

Квалификация

*магистр*

Фонд оценочных средств предназначен для контроля знаний обучающихся по направлению подготовки 09.04.04 – РФ, 710400 - КР «Программная инженерия» по дисциплине *« Процессы управления качеством программного обеспечения»*.

Фонд оценочных средств рассмотрен и утвержден
на заседании Высшей школы Цифровых технологий протокол №1 от 02.09.26

Директор Высшей школы Цифровых технологий:

к.т.н., доцентХмелева И.В.

Руководитель образовательной программы:

д.т.н., проф. Лыченко Н.М.

*Исполнители (разработчики):*
*к.т.н., доцент Мусина И.Р.*

*ст. преп., Гайдамако В.В*

---

СОГЛАСОВАНО:
Зам. директора по учебной работе
Политехнического института КРСУ Комарцов Н.М.

---

**Раздел 1. Перечень компетенций, с указанием этапов их формирования в процессе освоения дисциплины/практики**

| Формируемые компетенции | Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций | Виды оценочных средств/ шифр раздела в данном документе |
|---|---|---|
| **ОПК-8: Способен осуществлять эффективное управление разработкой программных средств и проектов.** | **Знать:**<br>- базовые научные положения и принципы программной инженерии сопровождения и эволюции ПС;<br>- научные основы различных методов адаптации, модификации и тестирования ПО;<br>-специальные программные продукты, реализующие эти методы и технологии. | **Блок А –** задания репродуктивного уровня<br><br>- Устный опрос |
| | **Уметь:**<br>- собирать такую информацию о работе функционирующего ПО, которая позволяет определить конкретные задачи для его модификации и/или адаптации с учётом изменившихся условий эксплуатации;<br>- применять адекватные методы и средства для реализации прикладных задач эволюции ПО в конкретной предметной области | **Блок В** – задания реконструктивного уровня<br><br>- Практические задания |
| | **Владеть:**<br>- навыками самостоятельного анализа работы сопровождаемого ПО и формулировки конкретных задач по его модификации и адаптации (дальнейшей эволюции ПО);<br>- методами и средствами для выполнения необходимых процессов и работ по разработке, тестированию и сопровождению программных систем;<br>- методами и средствами оценки качества ПО на всех этапах его разработки и сопровождения. | **Блок С** – задания практико-ориентированного и/или исследовательского уровня<br><br>- Практические задания<br>- Задания на самостоятельную работу |

**Раздел 2. Технологическая карта дисциплины**
**Процессы управления качеством программного обеспечения**

**Курс 1, семестр 2, Количество ЗЕ -3, Отчетность – экзамен**

| **Название модулей дисциплины согласно РПД** | **Контроль** | **Форма контроля** | **зачетный минимум** | **зачетный максимум** | **график контроля** |
|---|---|---|---|---|---|
| Модуль 1 | | | | | |
| Введение в процессы разработки качественного ПО. Оценка качества программ. | Текущий контроль | Практические работы №1-№3 | 4 | 8 | 5 |
| | Рубежный контроль | Самостоятельные работы №1-№3 | 9 | 15 | |
| Модуль 2 | | | | | |
| Современные методологии создания ПО. Верификация программ. | Текущий контроль | Практические работы № 4-№7 | 4 | 8 | 11 |
| | Рубежный контроль | Самостоятельные работы №4-№7 | 10 | 16 | |
| Модуль 3 | | | | | |
| Техники тестирования. Документирование результатов | Текущий контроль | Практические работы №8-№10 | 4 | 8 | 16 |
| | Рубежный контроль | Самостоятельные работы №8 - №10 | 9 | 15 | |
| ВСЕГО за семестр | | | 40 | 70 | |
| Промежуточный контроль (Экзамен) | | Письменная работа | 20 | 30 | |
| Семестровый рейтинг по дисциплине | | | 60 | 100 | |

**Раздел 3. Типовые контрольные задания и иные материалы, необходимые для оценки планируемых результатов обучения по дисциплине (оценочные средства)**

**Блок А**

*Вопросы для опроса*

*Модуль 1*

1. Определение качества программного обеспечения.
2. Процессы, необходимые для создания качественного ПО
3. Методологии создания ПО
4. Принципы Agile разработки ПО.
5. Модели жизненного цикла ПО.
6. Стандарты качества ПО.
7. Метрики для оценки атрибутов качества.
8. Задачи обеспечения качества.
9. Методы контроля (повышения) качества ПО.
8. Требования к ПО. Задачи извлечения и анализа требований. Необходимые свойства требований.
9. Требования к надежности.
10. Требования к эффективности.

11. Требования к удобству сопровождения.
13 Требования к переносимости (мобильности).
14. Причины недостаточного качества.

*Модуль 2*

1. Что такое верификация ПС?
2. Что такое валидация (аттестация) ПС?
3. Методы верификации ПО.
4. Основные принципы организации процесса тестирования.
5. Планирование верификации и аттестации: V – модель планирования разработки ПО.
6. Уровни тестирования.
7. Схема двухэтапного процесса тестирования.
8. Структура плана испытаний ПО, основные компоненты плана.
9. Что такое тестовые сценарии и как их составлять?
10. Какие существуют методики отбора тестовых сценариев?
11. Как провести контроль качества наборов тестов?
12. Взаимодействие в команде при тестировании.
13. Управление рисками.
14. Классификация видов тестирования.
15. Инструменты и инфраструктура при тестировании .
16. Контроль качества наборов тестов.
17. В чем различие между методиками тестирования («черный ящик», «серый ящик», «белый ящик»
18. Модульное тестирование. Как создается Unit –тест?
19. Структурное тестирование: тестирование потока управления, потока данных, тестирование ветвей, тестирование циклов
20. Тестирование сборки. Виды тестирования сборки.
21. Функциональное тестирование.
22. Системное тестирование.
23. Что входит в тестирование БД?
24. Как проводить отладку программы?

*Модуль 3*

1. Мутационное тестирование.
2. Что такое «мутант» и как его создать?
3. Тестирование производительности. Что входит в это понятие?
4. Дымовое тестирование (smoke testing).
5. Тестирование новой функции (new feature testing).
6. Подтверждающее тестирование.
7. Регрессионное тестирование.
8. Интуитивное тестирование (ad hoc testing).
9. Тестирование back-to back.
10. Негативное и позитивное тестирование.
11. Критерии завершения тестирования.
12. Что дефект программы?
13. Виды дефектов.
14. Что такое процесс управления дефектами?
15. Рекомендации по описанию дефектов.
16. Степени критичности дефектов.
17. Багтрекинговые системы.
18. Как измерить и оценить качество выполнения теста?

Тест (один из вариантов)

1. Процесс выполнения программы с целью обнаружения ошибок это – ...

оценивание
отладка
тестирование
структурирование

2. Что не обеспечивает тестирование?

Показ отсутствия дефектов
Обнаружение ошибок
Демонстрацию соответствия функций программы ее назначению
Отображение надежности как индикатора качества программы

3. Тестирование по методике «прозрачного» (белого) ящика это –

Поведенческое тестирование
Структурное тестирование
Негативное тестирование
Функциональное тестирование

4. Какого вида тестирования не существует?

Тестирование Back-To-Back
Системное тестирование;
Прогрессивное тестирование;
Тестирование безопасности;

5. При этом подходе тестирования интеграции, сборка и тестирование системы начинаются с модулей-атомов, располагаемых на нижних уровнях иерархии.

Восходящее тестирование интеграции
Нисходящее тестирование интеграции
Тестирование правильности интеграции
Сэндвич-тестирование

6. Проверка ПО на способность выполнять набор функций, удовлетворяющих требованиям заказчика - это…

Валидация
Верификация
Инспектирование
Консолидация

7. Что проверяет динамическая верификация системы?

Соответствие разработанной системы техническому заданию
Соответствие системы пожеланиям заказчика
Соответствие разработанных моделей проектирования техническому заданию
Корректность сформированных требований

8. Валидация системы — это

Проверка того, что мы разработали ПО согласно пожеланиям заказчика +
Проверка того, что мы разработали систему согласно ТЗ
Поверка того, что спецификация требований разработана правильно
Поверка того, что в коде нет синтаксических ошибок

9. Какое тестирование спрашивает: «как сильно можно расшатать систему, прежде чем она откажет?»

Тестирование безопасности
Стрессовое тестирование
Системное тестирование
Нагрузочное тестирование

10. В каком в хронологическом порядке этапы процесса тестирования расположены верно:

Тестирование компонентов, Тестирование модулей (юнитов), Тестирование системы, Приемочные испытания
Тестирование модулей(юнитов), Тестирование компонентов , Тестирование системы, Приемочные испытания

Тестирование системы, Тестирование компонентов, Тестирование модулей (юнитов), Приемочные испытания
Тестирование системы, Тестирование модулей (юнитов), Тестирование компонентов, Приемочные испытания

11. Что такое исчерпывающее тестирование?

Тестирование функциональности, с использованием неверных и верных данных ввода и входных условий
Тестирование функциональности, с использованием верных данных ввода и входных условий
Тестирование функциональности, с использованием неверных данных ввода и входных условий
Тестирование функциональности с конечным, продуманным множеством входных данных.

12. Что такое юзабилити-тестирование?

Проверка ПО на предмет того, насколько легко конечные пользователи способны понять и управлять приложением
Проверка того, насколько юзер готов к использованию программы.
Проверка корректности работы всех USE Case
Проверка способности программы к модернизации

13. Эта характеристика модели качества при использовании отображает точность и полноту, с которой пользователи достигают определенных целей.

Результативность (effectiveness)
Производительность (efficiency)
Удовлетворенность (satisfaction)
Свобода от риска (freedom from risk)

14. Характеристика системы, отображающая степень простоты эффективного и рационального переноса системы, продукта или компонента из одной среды (аппаратных средств, программного обеспечения, операционных условий или условий использования) в другую – это…

Мобильность
Сопровождаемость
Надежность
Защищенность

15. Характеристика системы, обеспечиваемая путем ограничения доступа людей, других продуктов или систем к данным в соответствии с типами и уровнями авторизации – это….

Мобильность
Сопровождаемость
Надежность
Защищенность

16. Требования, которые выражают цель, ради которой разрабатывается продукт (зачем он нужен, какая от него ожидается польза).

Бизнес-требования
Пользовательские требования
Функциональные требования
Нефункциональные требования

17. Требования, которые описывают свойства системы (удобство использования, безопасность, надёжность, расширяемость и т.д.), которыми она должна обладать при реализации своего поведения.

Бизнес-требования
Пользовательские требования
Функциональные требования
Нефункциональные требования

18. Тестирование — это проверка ….

Соответствия ПО требованиям
Проверка ТЗ на полноту и корректность
Проверка кода на соответствие алгоритмам
Проверка кода на «чистоту»

19. Какой вид тестирования не относится к тестированию производительности?
Стрессовое тестирование
Объемное тестирование
Нагрузочное тестирование
Тестирование восстанавливаемости

20. Какой степени критичности дефектов программы не существует?
Critical (критический)
Major (значительный)
Average (средней значимости)
Null (нулевая)

## Блоки B и C

**Практические задания**

### *Практическая работа 1. Оценка атрибутов качества*

Упражнение №1. Имеется web-приложение, расположенного по адресу http://www.klerk.ru. Для данного приложения дайте оценку следующим характеристикам качества: удобство использования, планировка и дизайн; соответствие содержания назначению, графика, цвета, оформление текста, устойчивость к ошибкам, независимость от платформы и особенности ОС. Результаты представить в виде распечатанного отчета.

Упражнение №2. Проанализируйте качество сайта автоматически с помощью онлайн программы. Можно использовать Sitechecker.pro - https://sitechecker.pro/ru/. Подготовьте отчет.

Упражнение №3. Определите, что такое качественный продукт/услуга в организации, в которой вы работаете. Подумайте над стоимостью плохого качества в вашей компании. Что есть плохое качество в вашей компании, как оно растет? Какие действия могут предотвратить плохое качество? Как в компании производится оценка качества продукта/услуги на соответствие стандартам качества? Уменьшается ли стоимость исправления ошибок путем увеличения затрат на предупреждение и оценку качества? Как повысить качество ПО в вашей компании?

### *Практическая работа 2. Оценка производительности и профилирование программы.*

Упражнение №1. В качестве исследуемой программы возьмите любое приложение Проведите анализ производительности в выбранной IDE, выявите «горячие» пути. Выясните причины «узких» мест программы.

Упражнение №2. Проведите профилирование программы, оцените требуемую память. Составьте отчет по результатам проведенного исследования.

### *Практическая работа 3. Расчет метрик кода в интегрированной среде программирования и их интерпретация.*

Упражнение №1. Выберите программу с открытым кодом. В интегрированной среде программирования (можно использовать MS VS любой версии, Net Beans, Eclipse и т.д.) оцените метрики кода: цикломатическую сложность, глубину наследования, связанность модулей (классов), количество строк кода, индекс удобства поддержки. Проведите анализ результатов. Представьте отчет со скриншотами.

Упражнение №2. Постройте потоковый граф для программы, выбранной в упражнении №1. Рассчитайте по формулам цикломатическую сложность программы. Сравните полученный вручную результат с метрикой, выданной автоматически в IDE.

***Практическая работа 4. Разработка плана тестирования с использованием шаблонов.***

Упражнение №1. Выберите проект с разработанным на него спецификацией требований. Протестировать спецификацию методом просмотра на предмет соответствия критериям качества требований.

Упражнение №2. По заданному шаблону разработайте план тестирования для разрабатываемой программы. Оформите отчет.

***Практическая работа 5. Разработка тестовых сценариев.***

Упражнение №1. Выбрать проект с имеющимся техническим заданием. Создать тестовые сценарии (включая тестовые варианты), используя для формирования данных методы разбиения по областям эквивалентности и граничных значений. Составить отчет с выводом об эффективности выбранных сценариев.

***Практическая работа 6. Структурное тестирование. Создание Unit- тестов.***

Упражнение №1. Разработайте программу по техническому заданию из упражнения №5 (можно реализовать программу на любом языке программирования и в любой IDE). По разработанным тестовым вариантам, создайте Unit – тесты. Проведите модульное тестирование в выбранной интегрированной среде программирования. Составьте отчет с выводом об эффективности проведенного модульного тестирования.

***Практическая работа №7. Тестирование базы данных***

Упражнение №1. Выберите или постройте самостоятельно базу данных (желательно на SQL SERVER). Протестируйте триггеры, хранимые процедуры. Составьте отчет по результатам тестирования.

***Практическая работа №8. Нагрузочное и стрессовое тестирование.***

Упражнение №1. В качестве исследуемой программы возьмите то же приложение, что и в практической работе №5. Проведите нагрузочное тестирования с помощью Apache JMeter или любой программы, позволяющей сделать это автоматически.

Для нагрузки исследуемого приложения задайте несколько различных значений количества запросов, включая пиковое (для стрессового тестирования). Скажем, 100,200,300,400,….1000. Для каждого значения определите время выполнения запроса, количество ошибок. Составьте отчет с выводом по полученным характерстикам.

Упражнение №2. Проведите профилирование программы, оцените требуемую память. Составьте отчет по результатам проведенного исследования.

***Практическая работа №9. Оценивание покрытия кода тестами***

Упражнение №1. Выберите приложение для тестирования. Разработайте тестовые варианты. Проведите Unit-тестирование. С помощью готового инструментария в IDE оцените процент покрытия кода. Составить отчет.

***Практическая работа №10.Описание дефектов***

Упражнение №1. Выберите приложение для тестирования с разработанным заранее техническим заданием и планом тестирования. Проведите тестирование согласно плану. Опишите обнаруженные дефекты в виде таблицы с указанием местоположения и критичности.

**Блок D (промежуточный контроль)**

**Вопросы для проверки уровня обученности ЗНАТЬ**

1. Определение качества программного обеспечения.
2. Процессы, необходимые для создания качественного ПО
3. Методологии создания ПО
4. Принципы Agile разработки ПО.
5. Модели жизненного цикла ПО.
6. Какие имеются стандарты качества ПО? Необходимость следования стандартам.
7. Стандарт ISO 9126. 4.
8. Какие имеются метрики для оценки атрибутов качества?
9. Задачи обеспечения качества.
10. Методы контроля (повышения) качества ПО.
11. Какие бывают требования к ПО.?
12. Задачи извлечения и анализа требований.
13. Необходимые свойства требований к ПО
14. Виды требований к ПО.
15. Требования к надежности.
16. Требования к эффективности.
17. Требования к удобству сопровождения
18. Требования к удобству использования
19. Требования к переносимости (мобильности).
20. Причины недостаточного качества ПС.
21. Методы верификации ПО: экспертиза, статический анализ, динамический анализ, формальные методы, синтетические методы верификации..
22. Стандарты IEEE 830 и IEEE 1233.16.
23. Основные принципы организации процесса тестирования.
24. Планирование верификации и аттестации.
25. V – модель планирования разработки ПО. Схема двухэтапного процесса тестирования.
26. Структура плана испытаний ПО, основные компоненты плана.
27. Уровни тестирования.
28. Технологии тестирования.
29. Тестовые сценарии. Методика отбора тестовых сценариев.
30. Контроль качества наборов тестов.
31. Классификация дефектов.
32. Взаимодействие в команде при тестировании.
33. Управление рисками процесса тестирования.
34. Инструменты и инфраструктура при тестировании.

**Задачи/задания для проверки уровня обученности УМЕТЬ и ВЛАДЕТЬ**

1. Оцените атрибуты качества заданного разработанного ПО и выполните рекомендации по их улучшению.
2. Протестируйте разработанные спецификации требований к разрабатываемому ПО.
3. Определите метрики кода разработанного ПО (к примеру, реализация задачи поиска элемента в массиве методом бинарного поиска), используя опции в IDE. Выполните анализ результатов.
4. Разработайте потоковый граф для реализация задачи поиска элемента в массиве методом бинарного поиска.
5. Рассчитайте цикломатическую сложность для задачи поиска элемента в массиве методом бинарного поиска.
6. Составьте план верификации для заданного ПО (к примеру, для задачи поиска элемента в массиве методом бинарного поиска).
7. Разработайте тестовые варианты для реализации задачи поиска элемента в массиве методом бинарного поиска.

8. Разработайте Unit-тесты для проверки значения скалярной переменной на примере функции вычисления наибольшего общего делителя двух положительных целых чисел: int NOD(int a, int b).
9. Разработайте Unit-тесты для проверки значений элементов массива (списка) на примере нахождения корней нелинейного уравнения с заданной очностью.
10. Выполните тестирование триггеров и хранимых процедур БД, разработанную для реализации программы управления складом готовых товаров.
11. Выполните оценку производительности и профилирование для программы поиска элемента в массиве методом бинарного поиска.
12. Разработайте тестовые сценарии для заданного веб-приложения.
13. Выполните экспертную оценку сайта КРСУ.
14. Выполните оценку сайта КРСУ с использованием программы sitechecker.pro.
15. Выполните оценку покрытия кода в IDE, реализующего задачу поиска элемента в массиве методом бинарного поиска.
16. Протестируйте программу, реализующую задачу поиска элемента в массиве методом бинарного поиска, методом "черного ящика" и составьте отчет с подробным описанием дефектов

**Пример**

## ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ РУБЕЖНОГО КОНТРОЛЯ № ___

1. Вопросы для проверки уровня обученности ЗНАТЬ
   1.1. Что такое качественный продукт/услуга согласно стандартам качества.
   1.2. Что входит в модель качества при использовании
   1.3. Уровни тестирования

Задание для проверки уровня обученности УМЕТЬ и ВЛАДЕТЬ

Продемонстрируйте, как использовать SEO (к примеру, sitechecker.pro) для проверки качества сайта iais.krsu.edu.kg.

**Раздел 4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций**

**Описание показателей и критериев оценивания компетенций, описание шкал оценивания**

Применяемые оценочные средства:

- Сдача практических работ на практических занятиях в соответствии с технологической картой дисциплины (текущая и рубежная аттестация),
- Прохождение тестирования,
- Устный опрос.
- Письменный опрос по экзаменационным билетам (промежуточная аттестация-экзамен),
- Сдача самостоятельных работ.

Все виды оценочных средств оцениваются в соответствии со шкалами оценивания.

**Устный опрос** на практических занятиях по отдельным темам проводится в течение всего периода обучения дисциплине. Результаты опроса учитываются при оценивании практических работ.

**ШКАЛА ОЦЕНИВАНИЯ ПРАКТИЧЕСКИХ (ЛАБОРАТОРНЫХ) РАБОТ**
**(текущий/рубежный контроль)**

- 85-100 % - Демонстрирует полное понимание проблемы. Все требования, предъявляемые к заданию, выполнены.
- 70-84 % - Демонстрирует значительное понимание проблемы. Все требования, предъявляемые к заданию, выполнены.
- 60-69 % - Демонстрирует частичное понимание проблемы. Большинство требований, предъявляемых к заданию выполнены.
- 31-60 % - Демонстрирует небольшое понимание проблемы. Многие требования, предъявляемые к заданию не выполнены.
- 0-30 % - Демонстрирует непонимание проблемы и даже не было попытки решить задачу.

**ШКАЛА ОЦЕНИВАНИЯ КОНТРОЛЬНЫХ РАБОТ (рубежный контроль)**

- 85-100 % - Демонстрирует полное понимание проблемы. Все задания выполнены.
- 70-84 % - Демонстрирует значительное понимание проблемы. Все задания выполнены, но содержат некоторые неточности.
- 60-69 % - Демонстрирует частичное понимание проблемы. Большинство требований, предъявляемых к заданию, выполнены.
- 31-60 % - Демонстрирует небольшое понимание проблемы. Многие требования, предъявляемые к заданию, не выполнены.
- 0-30 % - Демонстрирует непонимание проблемы или нет ответа и даже не было попытки решить задачу.

**ШКАЛА ОЦЕНИВАНИЯ ПИСЬМЕННОГО ОПРОСА (промежуточный контроль – «ЗНАТЬ»)**

Отметкой **(9-10 баллов)** оценивается ответ, который показывает прочные знания стандартов качества ПО, о принципов организации процесса тестирования, всех известных метрик для оценки атрибутов качества, всех известных методов контроля (повышения) качества ПО, все требования к ПО.

Отметкой **(7-8 баллов)** оценивается ответ, обнаруживающий знание стандартов качества ПО, основных принципов организации процесса тестирования, основных метрик для оценки атрибутов качества, основных методов контроля (повышения) качества ПО, основные требования к ПО.

Однако допускается одну - две неточности в ответе.

Отметкой **(4-6 баллов)** оценивается ответ, свидетельствующий о знании стандартов качества ПО, принципов организации процесса тестирования, метрик для оценки атрибутов качества, о методах контроля качества ПО, о требованиях к ПО.

Допускается несколько ошибок в содержании ответа.

Отметкой **(2-3 балла)** оценивается ответ, обнаруживающий незнание знания стандартов качества ПО, принципов организации процесса тестирования, метрик для оценки атрибутов качества, методов контроля (повышения) качества ПО, требований к ПО.

Отмечается отсутствие логичности и последовательности в ответе. Допускаются серьезные ошибки в содержании ответа.

Отметкой **(0 -1 балл)** оценивается ответ, при котором студент демонстрирует непонимание проблемы или нет ответа.

**ШКАЛА ОЦЕНИВАНИЯ ПРАКТИЧЕСКИХ ЗАДАНИЙ**
**(промежуточный контроль – «УМЕТЬ и ВЛАДЕТЬ»)**

Отметкой **(17-20 баллов)** оценивается ответ, при котором студент демонстрирует способность оценить атрибуты качества ПО, рассчитать оценочные метрики ПО, разработать план

тестирования с использованием шаблонов, провести профилирование ПО с анализом результатов, создать Unit-тесты.

Демонстрирует полное понимание проблемы. Все требования, предъявляемые к заданию, выполнены.

Отметкой **(11-16 баллов)** оценивается ответ, при котором студент демонстрирует способность оценить основные атрибуты качества ПО, рассчитать основные оценочные метрики ПО, разработать план тестирования, провести профилирование ПО, создать Unit-тесты.

Демонстрирует значительное понимание проблемы. Большинство требований, предъявляемых к заданию, выполнены.

Отметкой **(4-10 баллов)** оценивается ответ, при котором студент в основном демонстрирует неспособность оценить атрибуты качества ПО, рассчитать основные оценочные метрики ПО, разработать план тестирования, провести профилирование ПО, создать Unit-тесты и демонстрирует неспособность правильно интерпретировать полученные результаты.

Демонстрирует частичное или небольшое понимание проблемы. Многие требования, предъявляемые к заданию, не выполнены.

Отметкой **(0 -3 балла)** оценивается ответ, при котором студент демонстрирует непонимание проблемы или нет ответа и даже не было попытки решить задачи.

## ШКАЛА ОЦЕНИВАНИЯ ПИСЬМЕННОГО ОПРОСА

(**промежуточный контроль** – «ЗНАТЬ»)

Отметкой (7-10- баллов) оценивается ответ, который показывает прочные знания теоретических основ дисциплины, понимание и правильное применение терминологии, правильные ответы на 75-100% вопросов

Отметкой (5-7 баллов) оценивается ответ, который показывает знание теоретических основ дисциплины, но неполное понимание и не всегда правильное применение терминологии, даны правильные ответы на 50-74% вопросов, в ответах допущено некоторое количество неточностей.

Отметкой (3-4 баллов) оценивается ответ, свидетельствующий о знакомстве с некоторыми теоретическими основами дисциплины. Даны правильные ответы на 25-49% вопросов, допущены неточности и ошибки.

Отметкой (2 балла) оценивается ответ, обнаруживающий незнание теоретических основ дисциплины. Отмечается отсутствие логичности и последовательности в ответе. Менее 25% правильных ответов. Допущены серьезные ошибки в содержании ответа.

Отметкой (0-1 балл) оценивается ответ, при котором студент демонстрирует непонимание поставленных вопросов, или нет ответа.

## ШКАЛА ОЦЕНИВАНИЯ ПРАКТИЧЕСКИХ ЗАДАНИЙ

(**промежуточный контроль** – «УМЕТЬ и ВЛАДЕТЬ»)

Отметкой (8-10 баллов) оценивается ответ, при котором студент правильно отвечает на поставленные вопросы, Демонстрирует полное понимание проблемы. Все требования, предъявляемые к заданию, выполнены.

Отметкой (5-7 баллов) оценивается ответ, при котором студент в основном правильно отвечает на поставленные вопросы. Демонстрирует значительное понимание проблемы. Большинство требований, предъявляемых к заданию, выполнены.

Отметкой (2-4 баллов) оценивается ответ, при котором студент в основном не правильно отвечает на поставленные вопросы. Демонстрирует частичное или небольшое понимание проблемы. Многие требования, предъявляемые к заданию, не выполнены.

Отметкой (0 -1 балл) оценивается ответ, при котором студент демонстрирует непонимание проблемы или нет ответа и даже не было попытки решить задачи.

В экзаменационный билет включены три теоретических вопроса и практическое задание, соответствующие содержанию формируемых компетенций. Зачет с оценкой проводится в письменной форме. На ответ и решение задачи студенту отводится 80 минут. За ответ на теоретические вопросы студент может получить максимально 15 баллов, за выполнение практических заданий - 15 баллов.

По итогам прохождения дисциплины и с учетом шкал оценивания все набранные в результате текущей, рубежной и промежуточной аттестаций баллы суммируются и выставляется оценка .

Перевод баллов в оценку:

85 - 100 баллов – «отлично»

70 - 84 баллов – «хорошо»

60 - 69 баллов – «удовлетворительно»

менее 60 баллов – «неудовлетворительно»

**Раздел 5. Методические указания для обучающегося по освоению дисциплины и выполнению контрольных заданий**

Текущий контроль осуществляется в течение семестра в виде защиты практических работ.

Для обучающихся разработано учебно-методическое пособие, которое содержит теоретические сведения по тестированию и оценке качества программного обеспечения, задания для практических работ и методические указания к их выполнению. Обучающие материалы представлены в электронной папке преподавателя (локальная сеть кафедры Информационных и вычислительных технологий КРСУ). Выполнение всех практических работ завершается оформлением отчета, в котором приводится теория по теме работы, отчеты в виде скриншотов результатов, таблиц и графиков.

Рубежный контроль осуществляется в виде сдачи самостоятельных работ. Вопросы по контрольным работам представлены в разделе РПД Фонд оценочных средств. Результаты самостоятельных работ предоставляются виде распечатанных отчетов.

Для допуска к экзамену студент в течении семестра должен сдать все практические и самостоятельные работы.

**5.1. Общие рекомендации к организации самостоятельной работы**

Изучение дисциплины следует начинать с проработки настоящей рабочей программы, особое внимание, уделяя целям из задачам, структуре и содержанию курса. Работа с конспектом лекций. Необходимо просмотреть конспект сразу после занятий, отметить материал конспекта лекций, который вызывает затруднение для понимания. Попытайтесь найти ответы на затруднительные вопросы, используя предлагаемую литературу. Если самостоятельно не удалось разобраться в материале, то необходимо сформулировать вопросы и обратиться на ближайшей лекции за помощью к преподавателю. Каждую неделю нужно отводить время для повторения, пройденного материала, проверяя свои знания, умения и навыки по контрольным вопросам и тестам.

**5.2. Подготовка к практическим занятиям**

Перед посещением практического занятия изучить теорию вопроса, предполагаемого к исследованию. Оформление отчётов должно производиться по представленному образцу после окончания работы непосредственно в аудитории. Для подготовки к защите отчёта следует проанализировать экспериментальные результаты, обобщать результаты исследований в виде выводов, подготовить ответы на вопросы.