

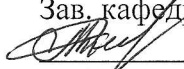
Государственное образовательное учреждение
«Приднестровский государственный университет им. Т.Г. Шевченко»

Рыбницкий филиал

Кафедра информатики и программной инженерии

«УТВЕРЖДАЮ»

Зав. кафедры-разработчика

 Л.А. Тягульская

протокол № 2 «21» сентября 2023 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ПО ДИСЦИПЛИНЕ

**Б1.О.16. «ТЕСТИРОВАНИЕ И ОТЛАДКА ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ»**

Направление:

2.09.03.04 «Программная инженерия»

Профиль:

«Разработка программно-информационных систем»

Квалификация:

бакалавр

Форма обучения:

очная, заочная

ГОД НАБОРА: 2020

Разработала: ст. преподаватель

 Гарбузняк Е.С.

« 21 » 09 20 23 г.

Рыбница, 2023

ПАСПОРТ
фонда оценочных средств по учебной дисциплине

1. В результате изучения дисциплины «Тестирование и отладка программного обеспечения» у обучающихся должны быть сформированы следующие компетенции:

Категория (группа) компетенций	Код и наименование	Код и наименование индикатора достижения универсальной компетенции
Общепрофессиональные компетенции и индикаторы их достижения		
	ОПК-4. Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью	ИД-2 _{ОПК-4.2.} Умеет применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы. ИД-3 _{ОПК-4.3.} Имеет навыки составления технической документации на различных этапах жизненного цикла информационной системы.
	ОПК-5. Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем	ИД-2 _{ОПК-5.2.} Умеет выполнять параметрическую настройку информационных и автоматизированных систем. ИД-3 _{ОПК-5.3.} Имеет навыки инсталляции программного и аппаратного обеспечения информационных и автоматизированных систем.
	ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ИД-2 _{ОПК-6.2.} Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ. ИД-3 _{ОПК-6.3.} Имеет навыки программирования, отладки и тестирования прототипов программно-технических комплексов задач.
Обязательные профессиональные компетенции и индикаторы их достижения		
	ПК-1. Владение навыками моделирования, анализа и использования формальных методов конструирования программного обеспечения	ИД-2 _{ПК-1.2.} Умеет использовать формальные методы конструирования программного обеспечения. ИД-3 _{ПК-1.3.} Владеет методами формализации и моделирования программного обеспечения.

2. Программа оценивания контролируемой компетенции:

№ п\п	Контролируемые модули, разделы (темы) дисциплины и их наименование	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
1.	ТЕКУЩИЙ КОНТРОЛЬ 1. Основные понятия тестирования и отладки ПО	ОПК-4, ОПК-5, ОПК-6, ПК-1 ИД-2 _{ОПК-4.2} ИД-3 _{ОПК-4.3} ИД-2 _{ОПК-5.2} ИД-3 _{ОПК-5.3}	Работа на лекциях Лабораторные работы Самостоятельная работа
	2. Виды тестирования и отладки ПО	ИД-2 _{ОПК-6.2} ИД-3 _{ОПК-6.3} ИД-2 _{ПК-1.2} ИД-3 _{ПК-1.3}	Работа на лекциях Лабораторные работы Самостоятельная работа Модульная контрольная
2.	ИТОГОВЫЙ КОНТРОЛЬ	ОПК-4, ОПК-5, ОПК-6, ПК-1	Тестирование Вопросы к экзамену

«УТВЕРЖДАЮ»
зав. кафедрой информатики
и программной инженерии,
доцент _____ Л.А. Тягульская
«21» _____ 09 _____ 2023 г.

**Вопросы и задания для текущего контроля
по дисциплине «Тестирование и отладка программного обеспечения»
для студентов IV курса
направления «Программная инженерия»
профиля «Разработка программно-информационных систем»,
8 семестр**

Модульная контрольная работа по дисциплине

**ЗАДАНИЕ № 1. ТЕОРЕТИЧЕСКИЙ ВОПРОС
(ДЛЯ СТУДЕНТОВ ЗАОЧНОЙ ФОРМЫ ОБУЧЕНИЯ)**

1. Необходимые и достаточные условия для проведения тестирования.
2. Модель качества ПО. Метрики по обеспечению качества ПО.
3. Критерии выбора тестов. Характеристики хорошего теста.
4. Способы тестирования базового пути и условий.
5. Тестирование ветвей и операций отношений.
6. Способы тестирования потоков данных и сценариев.
7. Особенности индустриального тестирования.
8. Жизненный цикл дефектов.
9. Автоматизация функционального тестирования: уровни автоматизации, архитектура тестов, стратегия использования тестов.
10. Особенности тестирования объектно-ориентированных модулей.
11. Тестирование объектно-ориентированной интеграции.
12. Проектирование объектно-ориентированных тестовых вариантов. Тестирование, основанное на ошибках и на сценариях.
13. Способы тестирования содержания класса. Стохастическое тестирование класса. Тестирование разбиений на уровне классов.
14. Автоматизация тестирования *web*-приложений.

ЗАДАНИЕ № 2. РАЗРАБОТКА ТРЕБОВАНИЙ К ПРОГРАММНОМУ ПРОДУКТУ

Постановка задачи

Разработать требования к программному продукту, разработанному ранее, например, в рамках курсовой работы по одной из дисциплин учебного плана.

Определение требований должно содержать:

Введение: цель создания документа, вид и область применения программного продукта.

1. Общее описание:

- 1.1. Функции ПП: работа в роли администратора (возможности для администрирования), работа в роли клиента (возможности для клиента).
- 1.2. Пользовательские характеристики (кто может быть пользователем описываемого ПП).
- 1.3. Общие ограничения: ограничения, связанные с оборудованием; разделение прав пользователей (регистрация пользователей); требования, накладываемые языками программирования.

- 1.4. Допущения и зависимости, связанные с ПП: операционные системы, браузеры, базы данных, зависимость от процессора.
2. Специальные требования:
- 2.1. Функциональные требования: пользовательский интерфейс, навигация (главное меню), аутентификация пользователя (роли пользователей и возможные действия), регистрация пользователя, просмотр информации о пользователях, изменение статуса пользователя, добавление нового пользователя, удаление пользователя из системы, редактирование данных пользователя.
- 2.2. Требования к производительности.

ЗАДАНИЕ № 3. МОДУЛЬНОЕ ТЕСТИРОВАНИЕ

Постановка задачи

1. Провести обзор разработанного программного кода. В случае обнаружения ошибок составить отчет следующего содержания (табл. 3.1):

Таблица 1.

Ошибки обзора				
Номер ошибки	Название модуля/функции	Описание ошибки	Важность ошибки (Высокая, Средняя, Низкая)	Ошибка исправлена (Да/Нет)

2. Для каждого модуля построить графы и вычислить цикломатические числа. Разработать тестовые случаи для каждого графа и представить их в таблице 2.

Таблица 2.

Модуль Main				
G	Номер сценария	Описание прохода	Контрольные примеры, позволяющие реализовать описание ситуации	Тест пройден (Да/Нет)

3. Провести модульное тестирование согласно составленным тестовым случаям. При необходимости разработать заглушки и привести их в таблице 3.

Таблица 3.

Описание заглушек			
Номер заглушки	Название модуля	Назначение заглушки	Тест пройден (Да/Нет)

- В случае не прохождения тестового случая составить отчет о каждой найденной ошибке в виде таблицы 4.

Таблица 4.

Модульное тестирование						
Номер ошибки	Название модуля	Описание прохода	Контрольные примеры, позволяющие реализовать описанную ситуацию	Описание ошибки	Важность ошибки	Ошибка исправлена (Да/Нет)

4. Построить схему взаимодействия модулей.
5. Разработать стратегию тестирования взаимодействия модулей и привести ее в виде таблицы

- 1.4. Допущения и зависимости, связанные с ПП: операционные системы, браузеры, базы данных, зависимость от процессора.
2. Специальные требования:
 - 2.1. Функциональные требования: пользовательский интерфейс, навигация (главное меню), аутентификация пользователя (роли пользователей и возможные действия), регистрация пользователя, просмотр информации о пользователях, изменение статуса пользователя, добавление нового пользователя, удаление пользователя из системы, редактирование данных пользователя.
 - 2.2. Требования к производительности.

ЗАДАНИЕ № 3. МОДУЛЬНОЕ ТЕСТИРОВАНИЕ

Постановка задачи

1. Провести обзор разработанного программного кода. В случае обнаружения ошибок составить отчет следующего содержания (табл. 3.1):

Таблица 1.

Ошибки обзора				
Номер ошибки	Название модуля/функции	Описание ошибки	Важность ошибки (Высокая, Средняя, Низкая)	Ошибка исправлена (Да/Нет)

2. Для каждого модуля построить графы и вычислить цикломатические числа. Разработать тестовые случаи для каждого графа и представить их в таблице 2.

Таблица 2.

Модуль Main				
G	Номер сценария	Описание прохода	Контрольные примеры, позволяющие реализовать описание ситуации	Тест пройден (Да/Нет)

3. Провести модульное тестирование согласно составленным тестовым случаям. При необходимости разработать заглушки и привести их в таблице 3.

Таблица 3.

Описание заглушек			
Номер заглушки	Название модуля	Назначение заглушки	Тест пройден (Да/Нет)

В случае не прохождения тестового случая составить отчет о каждой найденной ошибке в виде таблицы 4.

Таблица 4.

Модульное тестирование						
Номер ошибки	Название модуля	Описание прохода	Контрольные примеры, позволяющие реализовать описанную ситуацию	Описание ошибки	Важность ошибки	Ошибка исправлена (Да/Нет)

4. Построить схему взаимодействия модулей.
5. Разработать стратегию тестирования взаимодействия модулей и привести ее в виде таблицы

Таблица 5.

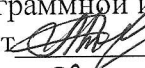
Взаимодействие модулей

Номер в последова- тельности	Описание последовательности	Контрольные примеры, позволяющие реализовать описанную ситуацию	Тест пройден (Да/Нет)

«УТВЕРЖДАЮ»

зав. кафедрой информатики

и программной инженерии,

доцент  Л.А. Тягульская

« 21 » 09 2023 г.

**Тестовые задания для итогового контроля
по итогам освоения дисциплины, а также для контроля самостоятельной работы студента
по дисциплине «Тестирование и отладка программного обеспечения»
для студентов IV курса
направления «Программная инженерия»
профиля «Разработка программно-информационных систем»,
8 семестр**

Указания: Напишите Вашу фамилию, номер группы и дату. Для ответа на вопрос с выбором варианта ответа достаточно написать номер вопроса и рядом букву, обозначающую правильный вариант из предложенных в тексте ответов на вопрос. Если Вы считаете правильными несколько вариантов ответов, то запишите через запятую соответствующие литеры букв.

Время: 45 мин

1. Подходы, используемые для обоснования истинности программ:
 - a) доказательство программы;
 - b) использование аналогий;
 - c) эксперимент над программой;
 - d) формальный и интерпретационный.
2. Методы анализа и локализации ошибки:
 - a) выполнение программы в уме;
 - b) метод контрольных точек и анализа трасс;
 - c) пошаговое выполнение.
3. Укажите, является ли программа аналогом математической формулы:
 - a) да;
 - b) нет;
 - c) является в отдельных случаях.
4. Недостатки функциональных критериев:
 - a) не проверяется соответствие со спецификацией;
 - b) не проверяются ошибки в структурных данных, требования к которым не зафиксированы в спецификации;
 - c) не проверяются ошибки, требования к которым не зафиксированы в спецификации.
5. Мощность множества тестов, формально необходимая для тестирования операции в машине с 32-разрядным машинным словом:
 - a) 4^9 ;
 - b) 2^{64} ;
 - c) 2^{32} .
6. Отметьте верные утверждения:
 - a) в фазу тестирования входят поиски и исправление ошибок;
 - b) отладка – процесс локализации и исправления ошибок;
 - c) тестирование – процесс поиска ошибок.
7. Log-файл используется для:

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

8. Укажите, возможно ли тестирование программы на всех допустимых значениях параметров:

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

9. К примитивам качества ПО относятся:

- a) функциональность;
- b) коммуникабельность;
- c) сопровождаемость;
- d) защищенность.

10. К критериям качества ПО относятся:

- a) функциональность;
- b) эффективность;
- c) модульность;
- d) защищенность.

11. Внешнее описание ПО включает:

- a) определение требований;
- b) спецификацию качества ПО;
- c) функциональную спецификацию ПО.

12. К методам контроля внешнего описания ПО относятся:

- a) постоянный контроль;
- b) пользовательский контроль;
- c) ручная имитация;
- d) ручная отладка.

13. К основным задачам отладки относятся:

- a) подготовить такой набор тестов и применить к ним ПО, чтобы обнаружить в нем по возможности большее число ошибок;
- b) найти ошибки;
- c) определить момент окончания отладки ПО или отдельной его компоненты.

14. Отладочные модули – это модули, управляющие:

- a) отладкой;
- b) тестируемой программой;
- c) интеграцией программы.

15. Наиболее эффективным является:

- a) тестирование модулей;
- b) комплексное тестирование;
- c) тестирование в реальных условиях;
- d) применение всех перечисленных методик тестирования.

16. К статическому тестированию модулей относятся:

- a) ручное тестирование;
- b) инспекции;
- c) сквозные просмотры;

d) тестирование структуры.

17. Метрический показатель сложности используется:

- a) при установлении проблемных областей;
- b) при создании контрольных примеров;
- c) при планировании тестирования;
- d) при создании потокового графа.

18. Способами проверки взаимодействия модулей являются:

- a) нисходящее тестирование;
- b) восходящее тестирование;
- c) монолитное тестирование;
- d) пошаговое тестирование.

19. Целями нагрузочного тестирования являются:

- a) оценка производительности и работоспособности приложения на этапе разработки и передачи в эксплуатацию;
- b) оптимизация производительности приложения, включая настройки серверов и оптимизацию кода;
- c) подбор соответствующей для данного приложения аппаратной (программной платформы) и конфигурации сервера.

20. Регрессионными могут быть:

- a) только функциональные тесты;
- b) только нефункциональные тесты;
- c) и функциональные тесты, и нефункциональные тесты.

21. Виды тестирования программного обеспечения, в зависимости от преследуемых целей, можно условно разделить на следующие группы:

- a) функциональные;
- b) нефункциональные;
- c) связанные с изменениями;
- d) все указанные виды.

22. К уровням тестирования относятся:

- a) приемочное тестирование;
- b) системное тестирование;
- c) интеграционное тестирование;
- d) объемное тестирование.

23. Этап процесса тестирования ПО, на котором проектируются и создаются тестовые случаи (тест кейсы), в соответствии с определенными ранее критериями качества и целями тестирования, называется:

- a) тестовым покрытием;
- b) тест дизайном;
- c) верификацией;
- d) валидацией.

24. Тестовый набор – это:

- a) комбинация тест скриптов, для проверки определенной части программного обеспечения, объединенной общей функциональностью или целями, преследуемыми запуском данного набора;
- b) набор инструкций, для автоматической проверки определенной части программного обеспечения;

d) тестирование структуры.

17. Метрический показатель сложности используется:

- a) при установлении проблемных областей;
- b) при создании контрольных примеров;
- c) при планировании тестирования;
- d) при создании потокового графа.

18. Способами проверки взаимодействия модулей являются:

- a) нисходящее тестирование;
- b) восходящее тестирование;
- c) монолитное тестирование;
- d) пошаговое тестирование.

19. Целями нагрузочного тестирования являются:

- a) оценка производительности и работоспособности приложения на этапе разработки и передачи в эксплуатацию;
- b) оптимизация производительности приложения, включая настройки серверов и оптимизацию кода;
- c) подбор соответствующей для данного приложения аппаратной (программной платформы) и конфигурации сервера.

20. Регрессионными могут быть:

- a) только функциональные тесты;
- b) только нефункциональные тесты;
- c) и функциональные тесты, и нефункциональные тесты.

21. Виды тестирования программного обеспечения, в зависимости от преследуемых целей, можно условно разделить на следующие группы:

- a) функциональные;
- b) нефункциональные;
- c) связанные с изменениями;
- d) все указанные виды.

22. К уровням тестирования относятся:

- a) приемочное тестирование;
- b) системное тестирование;
- c) интеграционное тестирование;
- d) объемное тестирование.

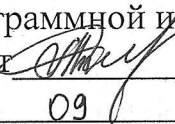
23. Этап процесса тестирования ПО, на котором проектируются и создаются тестовые случаи (тест кейсы), в соответствии с определенными ранее критериями качества и целями тестирования, называется:

- a) тестовым покрытием;
- b) тест дизайном;
- c) верификацией;
- d) валидацией.

24. Тестовый набор – это:

- a) комбинация тест скриптов, для проверки определенной части программного обеспечения, объединенной общей функциональностью или целями, преследуемыми запуском данного набора;
- b) набор инструкций, для автоматической проверки определенной части программного обеспечения;

«УТВЕРЖДАЮ»

зав. кафедрой информатики
и программной инженерии,
доцент  Л.А. Тягульская
« 21 » 09 2023 г.

**Вопросы к экзамену
по дисциплине «Тестирование и отладка программного обеспечения»
для студентов IV курса
направления «Программная инженерия»
профиля «Разработка программно-информационных систем»,
8 семестр**

1. Основные понятия отладки и тестирования программного обеспечения (ПО).
2. Аттестация ПО.
3. Внешнее описание ПО: понятие, назначение, роль в обеспечении качества ПО.
4. Определение требований к ПО.
5. Спецификация качества ПО.
6. Основные примитивы качества ПО.
7. Функциональная спецификация ПО.
8. Методы контроля внешнего описания ПО.
9. Различные подходы к тестированию: методы черного и белого ящиков, их комбинация.
10. Показатели эффективности различных методов тестирования.
11. Группы тестирования ПО (в зависимости от преследуемых целей).
12. Функциональные виды тестирования.
13. Нефункциональные виды тестирования.
14. Виды тестирования, связанные с изменениями.
15. Уровни тестирования.
16. Критерии выбора тестов: структурные, функциональные, стохастические, мутационный.
17. Модульное тестирование: понятие, задачи, виды тестирования модулей, используемые методы.
18. Тестирование структуры программных модулей и взаимодействия модулей.
19. Объектно-ориентированное тестирование: тестирование классов, функциональных возможностей и взаимодействия объектов, оцениваемые факторы.
20. Отладка ПО: задачи, виды, принципы отладки ПО.
21. Надежность ПО.
22. Рекомендации по организации отладки ПО.
23. Автономная отладка ПО: отладочные модули, интеграция программы.
24. Нисходящее и восходящее тестирование, их комбинация.
25. Комплексная отладка ПО: тестирование архитектуры, внешних функций, качества, документации по применению, определения требований к ПО.
26. Регрессионное тестирование: цели, задачи, виды.
27. Тестовое покрытие. Построение тестовых наборов.

28. Поточковый граф для модуля. Цикломатическое число.
29. Отладочные имитаторы (заглушки).
30. Интеграционное тестирование.
31. Автоматизированное тестирование ПО.
32. Инструменты для нагрузочного тестирования.
33. Инструменты для автоматизированного функционального тестирования.

Экзаменатор, ст. преподаватель Е.С. Гарбузняк