

Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Радиофизический факультет

УТВЕРЖДЕНО:

Декан

А. Г. Коротаев

Оценочные материалы по дисциплине

Верификация и тестирование программного обеспечения

по направлению подготовки

03.04.03 Радиофизика

Направленность (профиль) подготовки:

Радиофизика, электроника и информационные системы

Форма обучения

Очная

Квалификация

Магистр

Год приема

2025

СОГЛАСОВАНО:

Руководитель ОП

Д.Я. Суханов

Председатель УМК

А.П. Коханенко

Томск – 2025

1. Компетенции и индикаторы их достижения, проверяемые данными оценочными материалами

Целью освоения дисциплины является формирование следующих компетенций:

ОПК-1 Способен применять фундаментальные знания в области физики и радиофизики для решения научно-исследовательских задач, в том числе в сфере педагогической деятельности;

ОПК-3 Способен применять современные информационные технологии, использовать компьютерные сети и программные продукты для решения задач профессиональной деятельности.

ПК-1 Способен производить анализ состояния научно-технической проблемы, технического задания, формулировать цель и задачи научного исследования в области радиофизики и электроники.

ПК-2 Способен осуществлять построение математических моделей объектов исследования и выбор готового или разработку нового алгоритма решения задачи.

ПК-3 Способен использовать современное оборудование для решения профессиональных задач в области радиофизики и электроники.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИОПК 1.1 Представляет современную научную картину мира, выявляет естественнонаучную сущность проблемы, формулирует задачи в области радиофизики и радиоэлектроники и определяет пути их решения

ИОПК 1.2 Организует проведение научного исследования и разработку в области радиофизики и радиоэлектроники

ИОПК 3.1 Осуществляет поиск научно-технической информации с использованием информационных технологий

ИОПК 3.2 Предлагает новые идеи и подходы к решению научно-исследовательских и прикладных задач с использованием информационных систем и технологий

ИПК 1.1 Формулирует проблему и определяет предметную область исследования

ИПК 1.2 Проводит поиск и анализ научно-технической информации и патентной документации, отечественного и зарубежного опыта в выбранной области радиофизики и электроники

ИПК 1.3 Представляет информацию в систематизированном виде, формулирует цель исследования

ИПК 2.1 Формулирует постановку задачи, определяет параметры и функции разрабатываемой системы

ИПК 2.2 Определяет алгоритм и набор параметров, с учётом которых должно быть проведено моделирование устройства или системы

ИПК 2.3 Проводит компьютерное моделирование устройства или системы

ИПК 3.1 Понимает принципы действия устройств и систем, предназначенных для решения задач в области радиофизики и электроники

ИПК 3.2 Проводит измерения с использованием современных устройств и систем для решения профессиональных задач

2. Оценочные материалы текущего контроля и критерии оценивания

Элементы текущего контроля:

- тесты;
- опрос;
- проверка результатов проделанной лабораторной работы.

Контрольные вопросы по дисциплине (ИОПК 1.1, ИОПК 1.2, ИОПК 3.1, ИОПК 3.2, ИПК 1.1, ИПК 1.2, ИПК 1.3):

Этапы разработки ПО.

- а) Каковы основные этапы разработки ПО?
- б) Для чего нужна верификация ПО?
- в) Для чего нужно тестирование ПО?
- г) Понятие качества ПО.
- д) Понятие жизненного цикла ПО.
- е) Модели жизненного цикла.
- ж) Верификация и тестирование в различных моделях жизненного цикла ПО.

Верификация ПО.

- а) Основные подходы к верификации.
- б) В чём разница между статической и динамической верификацией?

Верификаторы SPIN и BLAST.

- а) Какой подход к верификации используется в верификаторе SPIN?
- б) Основные утилиты верификатора SPIN.
- в) Какой подход к верификации используется в верификаторе BLAST?
- г) Основные утилиты верификатора BLAST.

Виды тестирования ПО. Подходы к тестированию ПО.

- а) Привести примеры различных видов тестирования (функциональное тестирование, нагрузочное тестирование, тестирование взаимодействия, тестирование устойчивости) и кратко описать каждый из них.
- б) В чём сходство и различие между «белым ящиком» и «чёрным ящиком»?

Тестирование на основе случайных данных.

- а) В чём состоит идея тестирования на основе случайных данных?
- б) Как можно улучшить качество случайных тестов?

Метод разбиения области определения входных данных.

- а) Что в тестировании понимается под доменами области определения входных данных?
- б) Как можно построить домены?
- в) При известных доменах какие значения лучше выбирать в качестве тестовых данных? Почему?

Покрытие программного кода.

- а) Кратко охарактеризовать различные виды покрытия программного кода (покрытие строк, покрытие ветвей условных операторов, покрытие логических условий).
- б) Объяснить, как покрытие программного кода может быть использовано для оценки качества тестов ПО.
- в) Инструмент Gcov.

Особенности тестирования, связанные с недостижимыми участками кода.

- а) Понятие недостижимости участка кода.
- б) В чём разница между логической недостижимостью участка кода и фактической недостижимостью участка кода? Как это отразится на тестировании?

Тестирование на основе формальных моделей.

- а) Кратко описать формальные модели, используемые при тестировании ПО (автоматы, полуавтоматы, блок-схемы).
- б) Как строятся формальные модели ПО?
- в) Форматы описания моделей.
- г) Различные архитектуры тестирования.
- д) Понятие полноты тестирования.
- е) Способы обеспечения гарантированной полноты тестирования (область неисправности, цели тестирования).
- ж) Отношения между различными моделями.
- з) Что понимается под термином «тестирование на основе формальных моделей»?

Применение систем UniTESK и TorX для тестирования ПО.

- а) Драйверы, заглушки, архитектура тестирования в UniTESK и TorX.
- б) Организация процесса тестирования в UniTESK и TorX.

Утечка ресурсов.

- а) Понятие утечки ресурсов. Возникающие при этом проблемы.
- б) Инструмент Valgrind.

Критерии оценивания: тест считается пройденным, если обучающий ответил правильно как минимум на половину вопросов.

Темы и задания для лабораторных занятий (ИПК 2.1, ИПК 2.2, ИПК 2.3, ИПК 3.1, ИПК 3.2)

1. Статическая и динамическая верификация ПО.
При выполнении данной лабораторной работы студент должен реализовать программу расчёта процентов при снятии и зачислении средств на счёт в банке на языке программирования Си. Выбрать метод статической или динамической верификации и обосновать свой выбор. Провести верификацию программной реализации (вручную) согласно выбранному методу. Объяснить результаты верификации. Оформить отчёт о проделанной лабораторной работе.
2. Верификатор SPIN.
При выполнении данной лабораторной работы студент должен изучить основные утилиты верификатора SPIN. Провести верификацию программной реализации из лабораторной работы 1 с помощью верификатора SPIN. Объяснить результаты верификации, а также сравнить результаты с результатами лабораторной работы. Оформить отчёт о проделанной лабораторной работе.
3. Верификатор BLAST.
При выполнении данной лабораторной работы студент должен изучить основные утилиты верификатора BLAST. Провести верификацию программной реализации из лабораторной работы 1 с помощью верификатора BLAST.

Объяснить результаты верификации, а также сравнить результаты с результатами лабораторных работ 1 и 2. Оформить отчёт о проделанной лабораторной работе.

4. Средство описания тестов JUnit для программ на языке Java.

При выполнении данной лабораторной работы студент должен изучить инструмент описания тестов JUnit, подключённого к среде разработки NetBeans (или eclipse, по выбору студента). Реализовать программу расчёта процентов при снятии и зачислении средств на счёт в банке на языке программирования Java в среде разработки NetBeans (или eclipse, по выбору студента). Пользуясь JUnit подготовить класс тестов (пустой) для этой программной реализации. Оформить отчёт о проделанной лабораторной работе.

5. Тестирование на случайных данных.

Тестирование на основе разбиения области определения. При выполнении данной лабораторной работы студент, опираясь на результаты лабораторной работы 4, должен заполнить тестовый класс случайными данными и провести тестирование программной реализации. Разбить на домены область входных данных и, пользуясь ими, вновь заполнить тестовый класс и провести повторное тестирование. Сравнить результаты тестирования в первом и во втором случае. Оформить отчёт о проделанной лабораторной работе.

6. Покрывание программного кода. Оценка качества тестов.

При выполнении данной лабораторной работы студент должен включить в среде разработки NetBeans (или eclipse, по выбору студента) инструмент, оценивающий покрытие кода и, пользуясь тестовыми классами из лабораторной работы 5, провести тестирование программной реализации из лабораторной работы 4, измерить покрытие программного кода, сделать вывод о качестве тестов. Оформить отчёт о проделанной лабораторной работе.

7. Построение формальной модели ПО.

При выполнении данной лабораторной работы студент должен выбрать одну из формальных моделей, обосновать свой выбор и в рамках этой модели описать программную реализацию из лабораторной работы 4. Оформить отчёт о проделанной лабораторной работе.

8. Тестирование с помощью инструмента TorX.

При выполнении данной лабораторной работы студент должен изучить основные утилиты и способ организации тестирования в инструменте TorX. Программно реализовать протокол SCP, построить полуавтоматную модель данного протокола. Организовать тестирование программной реализации SCP относительно полученной полуавтоматной модели в системе TorX. Оценить результаты тестирования. Оформить отчёт о проделанной лабораторной работе.

9. Тестирование с помощью инструмента UniTESK.

При выполнении данной лабораторной работы студент должен изучить основные утилиты и способ организации тестирования в инструменте UniTESK. Пользуясь программной реализацией протокола SCP и полуавтоматной моделью данного протокола из лабораторной работы 9, организовать тестирование в системе UniTESK. Оценить результаты тестирования и сравнить их с результатами лабораторной работы 9. Оформить отчёт о проделанной лабораторной работе.

Текущая аттестация по лабораторным работам проводится в виде отчетов по лабораторной работе. Результаты выполнения лабораторных работ определяются оценками «зачет», «незачет».

3. Оценочные материалы итогового контроля (промежуточной аттестации) и критерии оценивания

Каждый билет для устного экзамена состоит из двух теоретических вопросов по двум темам дисциплины. В качестве дополнительных вопросов на устном экзамене используются контрольные вопросы.

Список вопросов к экзамену по дисциплине (ИОПК 1.1, ИОПК 3.2, ИПК 1.1, ИПК 1.3, ИПК 2.1)

1. Этапы разработки ПО.
2. Жизненный цикл ПО. Виды моделей жизненных циклов ПО.
3. Верификация и тестирование ПО. Основные отличия верификации и тестирования.
4. Основные подходы к верификации ПО.
5. Верификатор SPIN.
6. Верификатор BLAST.
7. Инструмент Gcov.
8. Инструмент Valgrind.
9. Подходы к тестированию ПО.
10. Модели «чёрный ящик» и «белый ящик».
11. Случайная генерация входных данных. Целевая функция.
12. Метод тестирования на основе разбиения области определения входных данных.
13. Понятие покрытия программного кода. Виды покрытий программного кода.
14. Понятие логически недостижимого участка кода.
15. Понятие фактически недостижимого участка кода.
16. Особенности тестирования, связанные с недостижимыми участками кода.
17. Формальные модели поведения систем.
18. Блок-схема программного кода.
19. Отношения между различными формальными моделями.
20. Понятие полноты тестирования.

Результаты экзамена определяются оценками «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Оценка «отлично» выставляется, если даны правильные и развернутые ответы на все вопросы билета и правильный ответ на дополнительный вопрос.

Оценка «хорошо» выставляется, если даны правильные и развернутые ответы на все вопросы билета, но дан неправильный ответ на дополнительный вопрос.

Оценка «удовлетворительно» выставляется, если дан неправильный ответ на 1 вопрос билета и дан правильный ответ на 1 дополнительный вопрос.

Оценка «неудовлетворительно» выставляется, если даны неправильные ответы на 2 вопроса билета, либо дан неправильный ответ на 1 вопрос билета и даны 2 неправильных ответа на дополнительные вопросы.

4. Оценочные материалы для проверки остаточных знаний (сформированности компетенций)

Тест

1. Для чего нужна верификация ПО? (ИОПК 1.1)
 - а) Для проверки соответствия требований конечным пользователям.
 - б) Для проверки корректности документации.
 - в) Для подтверждения соответствия программного обеспечения требованиям спецификации.
2. Какой подход к верификации используется в верификаторе SPIN? (ИПК 2.3)

- а) Тестирование методом черного ящика.
 - б) Проверка модели.
 - в) Тестирование методом белого ящика.
 - г) Симуляция работы системы.
3. Что в тестировании понимается под доменами области определения входных данных? (ИПК 1.1)
- а) Подмножества возможных значений входных данных, которые обрабатываются системой по-разному.
 - б) Описание всех возможных выходных данных системы.
 - в) Совокупность тестовых случаев для проверки корректности системы.
 - г) Варианты входных данных, которые не могут быть обработаны системой.
4. В чем заключается основной недостаток статического анализа кода? (ИПК 3.1)
- а) Занимает слишком много времени.
 - б) Не находит ошибки, связанные с исполнением программы.
 - в) Требуется использование сложного оборудования.
 - г) Невозможно автоматизировать.
5. В чем заключается основной принцип регрессионного тестирования? (ИПК 2.1)
- а) Проверка того, что изменения в коде не привели к появлению новых ошибок.
 - б) Полное переписывание тест-кейсов после каждого изменения кода.
 - в) Автоматизация всех возможных тестов.
 - г) Непрерывное выполнение тестов в реальном времени.
6. Что такое покрытие кода в контексте тестирования? (ИПК 3.2)
- а) Совокупность всех ошибок, обнаруженных в программе.
 - б) Метрика, показывающая, какая часть исходного кода была выполнена во время тестирования.
 - в) Способность программы обрабатывать различные виды входных данных.
 - г) Описание всех возможных тестовых случаев.
7. Какой из методов тестирования наиболее подходит для проверки поведения системы с большим количеством пользователей? (ИОПК 1.2)
- а) Юнит-тестирование.
 - б) Интеграционное тестирование.
 - в) Нагрузочное тестирование.
 - г) Статическое тестирование.

Ключи: 1 в) 2 б) 3 а) 4 б) 5 а) 6 б) 7 в)

Информация о разработчиках

Лапутенко Андрей Владимирович, к.т.н., кафедра информационных технологий в исследовании дискретных структур радиофизического факультета, доцент