**Программа учебной дисциплины** «**Обеспечение качества и тестирование ПО»**

Утверждена

Академическим советом ООП

Протокол № от «\_\_»\_\_\_\_\_20\_\_ г.

|  |  |
| --- | --- |
| Автор  | Кулямин В.В, к.ф.-м.н., kuliamin@ispras.ru |
| Число кредитов  | 5 |
| Контактная работа (час.)  | 64 |
| Самостоятельная работа (час.)  | 126 |
| Курс  | 3 |
| Формат изучения дисциплины | без использования онлайн курса |

1. **ЦЕЛЬ, РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ И ПРЕРЕКВИЗИТЫ**

Целями освоения дисциплины «Обеспечение качества и тестирование ПО» являются овладение студентами основными понятиями в области программной инженерии, связанными с методами обеспечения качества программного обеспечения (ПО), и навыками использования методов тестирования ПО с упором на использование формализуемых моделей поведения.

В результате освоения дисциплины студент должен:

**знать:**

- основные понятия в области обеспечения качества ПО;

- основные методы моделирования поведения ПО и ситуаций его функционирования;

- основные методы оценки полноты тестирования ПО;

- цели, задачи и основные методы разработки тестов и тестирования ПО.

**уметь:**

- разрабатывать и реализовывать тесты для программ различных типов;

- оценивать полноту тестирования программ различных типов;

- анализировать результаты выполнения тестов;

- применять формализованные модели при разработке тестов и оценке их полноты;

**владеть:**

- навыками решения типовых задач разработки тестов для ПО;

- навыками анализа результатов выполнения тестов;

- навыками применении формализованных моделей для разработки тестов.

Изучение дисциплины «Обеспечение качества и тестирование ПО» базируется на следующих дисциплинах:

- «Дискретная математика» в объеме двух первых курсов;

- «Программирование» в объеме двух первых курсов;

- «Алгоритмы и структуры данных» в объеме двух первых курсов;

- «Архитектура вычислительных систем» в объеме двух первых курсов.

Для освоения учебной дисциплины студенты должны владеть следующими знаниями и компетенциями:

* знать основные виды математических структур (логические исчисления, алгебраические структуры, конечные автоматы);
* обладать базовыми навыками описания и анализа математических структур;
* обладать базовыми навыками программирования и анализа программ.

Основные положения дисциплины должны быть использованы в дальнейшем при изучении следующих дисциплин:

1. Верификация программного обеспечения;
2. Формальные методы верификации и тестирования телекоммуникационных протоколов и сервисов.
3. **СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ**

**Тема 1. Качество программного обеспечения и методы его контроля.**

Качество ПО. Характеристики качества, стандарты ISO 9126 и ISO 25010. Задачи обеспечения качества ПО. Методы контроля качества ПО. Верификация и валидация. Методы верификации ПО: экспертиза, статический анализ, динамический анализ, формальные методы, синтетические методы верификации. Требования к ПО. Задачи извлечения и анализа требований. Необходимые свойства требований, стандарты IEEE 830 и IEEE 1233.

**Тема 2. Методы верификации ПО.**

Классификация методов верификации ПО. Экспертиза. Статический анализ. Верификационный мониторинг. Формальные методы верификации. Дедуктивный анализ. Проверка моделей. Примеры применения различных методов верификации.

**Тема 3. Цели и задачи тестирования ПО.**

Различные определения тестирования ПО. Тесты и тестовые наборы. Тестируемая система, тестируемая реализация, тестируемый интерфейс. Точки контроля и наблюдения. Роль требований. Модели поведения и модели ситуаций. Характеристики ситуаций. Цели тестирования в рамках жизненного цикла ПО. Задачи разработки тестов. Виды тестирования. Модульное, интеграционное и системное тестирование.

**Тема 4. Организация тестовых наборов.**

Тестовые варианты. Основные паттерны организации тестовых наборов. Квалификаторы тестов. Связи между тестами и другими артефактами разработки. Основные модули тестов. Тестовые заглушки. Тестовые оракулы. Способы построения оракулов. Виды инструментов разработки тестов и тестирования. Семейство инструментов xUnit.

**Тема 5. Модели поведения ПО.**

Логико-алгебраические и исполнимые модели. Абстрактные типы данных. Конечные автоматы. Расширенные конечные автоматы. Системы размеченных переходов. Использование разных видов моделей для описания функциональности.

**Тема 6. Модели ситуаций и критерии полноты тестирования.**

Общая структура моделей ситуаций. Критерии полноты и критерии покрытия. Доменные, функциональные, структурные и проблемные критерии полноты. Использование графов, грамматик и логических выражений для построения критериев полноты тестирования. Типовые критерии покрытия кода: покрытие инструкций, ветвлений, покрытия потоков данных, покрытие условий, комбинаций условий, модифицированное покрытие условий и ветвлений (MC/DC). Критерии покрытия на интеграционном и системном уровнях.

**Тема 7. Основные методы построения тестов. Вероятностные и нацеленные методы.**

Вероятностные, нацеленные и комбинаторные методы тестирования. Области применимости различных методов. Вероятностные методы. Распределение рисков. Марковские цепи. Нацеленные методы. Метод функциональных диаграмм. Техника выбора экстремальные значений. Доменное тестирование. Тестирование с помощью сценариев. Часто встречающиеся ошибки для различных типов данных.

**Тема 8. Комбинаторные методы построения тестов.**

Основные комбинаторные техники построения тестов. Дерево классификации. Использование грамматик. Покрывающие наборы. Алгоритмы построения покрывающих наборов.

**Тема 9. Автоматные методы построения тестов.**

Задача тестирование на соответствие конечному автомату. Методы построения тестов для конечных полных детерминированных автоматов. D-метод. W-метод. Wp-метод. Построение тестов для расширенных и недетерминированных автоматов. Различные виды соответствия реализации и спецификации.

**Тема 10. Интегрированные технологии построения тестов.**

Элементы технологии UniTESK. Использование программных контрактов для описания поведения. Редукция автомата по критерию полноты. Описание асинхронного поведения. Тестирование параллелизма и асинхронности на соответствие семантике линеарнизуемости набора событий.

1. **ОЦЕНИВАНИЕ**

# **Формы контроля знаний студентов**

|  |  |  |  |
| --- | --- | --- | --- |
| Тип контроля | Форма контроля | 3 год | Параметры  |
| 1 | 2 | 3 | 4 |
| Текущий(неделя) | Контрольная работа |  | 8 |  |  | Письменная работа 45 минут |
| Домашнее задание | 2,4,6  | 2,4,6 |  |  |  |
| Итоговый | Экзамен |  | 8 |  |  | Письменный экзамен 80 минут |

**Критерии оценки знаний, навыков**

Оценки по всем формам текущего контроля выставляются по 10-ти балльной шкале.

Контрольная работа состоит из 8 заданий, правильное решение которых оценивается от 2 до 6 технических баллов, в зависимости от их сложности. Контрольная работа в целом оценивается в сумме от 0 до 30 технических баллов. Экзамен состоит из 4 задач, правильное решение которых оценивается от 8 до 10 технических баллов. Экзамен в целом оценивается от 0 до 34 технических баллов. Процент суммарно набранных баллов от максимально возможного количества определяет оценку за контрольную работу и за письменный экзамен.

Домашние задания оцениваются по аналогичной схеме.

|  |  |
| --- | --- |
| Оценка | Набранные баллы (% от максимально возможных) |
| отлично (10) | более 90 |
| отлично (9) | от 80 до 90 включительно |
| отлично (8) | от 70 до 80 включительно |
| хорошо (7) | от 60 до 70 включительно |
| хорошо (6) | от 50 до 60 включительно |
| удовлетворительно (5) | от 40 до 50 включительно |
| удовлетворительно (4) | от 30 до 40 включительно |
| неудовлетворительно (3) | от 20 до 30 включительно |
| неудовлетворительно (2) | от 10 до 20 включительно |
| неудовлетворительно (1) | не более 10 |

**Порядок формирования оценок по дисциплине**

Преподаватель оценивает работу студентов на практических занятиях и выполненные ими домашние задания. Оценка за работу студента на практических занятиях и в рамках домашних заданий выставляется в рабочую ведомость. Полученная оценка по 10-ти балльной шкале определяется перед итоговым контролем - *Оауд*.

Преподаватель оценивает контрольную работу студентов. Оценка за контрольную работу студента выставляется в рабочую ведомость. Оценка по 10-ти балльной шкале за контрольную работу – *Оконтр. работа*.

Накопленная оценка за текущий контроль учитывает результаты студента по текущему контролю следующим образом:

О*накопл.* = *0,5*\* Оауд + *0,5*\*Оконтр.работа

В диплом выставляет результирующая оценка по учебной дисциплине, которая формируется по следующей формуле:

*Орезульт = 0,67*\**Онакопл* + *0,33*\**Оэкз*

Способ округления — арифметический.

1. **ПРИМЕРЫ ОЦЕНОЧНЫХ СРЕДСТВ**

# Оценочные средства для текущего контроля студента

**Примерный вариант письменной контрольной работы для текущего контроля успеваемости.**

1. К каким из характеристик качества по ISO 25010 относятся описанные требования? Перечислите номера соответствующих характеристик качества в первом столбце строки с описаниями требований в приведенной ниже таблице.

1. Функциональность
2. Надежность
3. Эффективность, производительность
4. Удобство использования
5. Переносимость
6. Удобство сопровождения
7. Совместимость
8. Защищенность

|  |  |
| --- | --- |
|  | Система должна работать 24 часа в сутки, 7 дней в неделю. Допустимый простой составляет 10 минут в год |
|  | Контроль работоспособности основных функций системы после внесения в нее изменений должен требовать не более одного человеко-дня |
|  | Система должна работать на платформах Windows 7, Windows 8 и Windows 10 |
|  | В проектной документации должна присутствовать полная схема зависимостей между модулями системы |
|  | Доступ пользователя к счетам должен быть невозможен без аутентификации  |
|  | Система должна быть способна обрабатывать до 3000 транзакций в секунду |

2. К каким из характеристик качества по ISO 25010 относятся описанные ошибки? Перечислите номера соответствующих характеристик качества (из предыдущего задания) в первом столбце строки с описанием ошибки в приведенной ниже таблице.

|  |  |
| --- | --- |
|  | Для выполнения наиболее частой операции — получения отчета за некоторый временной отрезок — требуется нажать не менее 8-ми кнопок на 4-х диалогах |
|  | Время сортировки списка пользователей возрастает примерно пропорционально кубу их количества |
|  | В систему можно войти под именем пользователя ‘admin’ и пустым паролем, этого пользователя нельзя удалить или заблокировать и сменить его пароль тоже нельзя |
|  | Внесение первого же изменения, которое было затребовано пользователями, привело к полной перестройке архитектуры системы |
|  | При отказе системы все данные о переводах денег, сделанных пользователем во время того сеанса, когда система отказала, пропадают |
|  | ПО неспособно работать с документами формата MS Word, хотя это заявлено в его спецификациях |

3. К какому виду относятся методы контроля качества, описанные в приведенных сценариях? Укажите соответствующий номер в первом столбце строки с описанием способа проверки в приведенной ниже таблице.

1. Экспертиза
2. Статический анализ
3. Формальная верификация
4. Тестирование
5. Мониторинг

|  |  |
| --- | --- |
|  | Группа, выполняющая проверку, формулирует набор сценариев возможных действий пользователей. Разработчики системы описывают работу системы в рамках этих сценариев и ее результаты на каждом шаге. Каждый проверяющий по ходу работы задает вопросы и составляет список замечаний и предложений по улучшению системы. После этого предложения и замечания совместно обсуждаются и сортируются по важности на три группы: критически важные, важные, несущественные. Замечания из первых двух групп принимаются к исправлению. |
|  | Код системы обрабатывается инструментом, который выдает список сообщений о возможных ошибках, включающих разыменование нулевых указателей, использование неинициализированных или уже освобожденных областей памяти, использование памяти за границами массивов и буферов. Эксперты просматривают список сообщений и код и выделяют те сообщения, которые соответствуют действительности. |
|  | Группа проверки получает требования к системе и на их основе формирует список сценариев работы системы, затрагивающих каждое проверяемое утверждение из требований. Каждый сценарий пополняется правильными результатами работы системы на каждом шаге. Все эти сценарии выполняются, собираются полученные от системы результаты, и проводится их анализ. |
|  | Инструмент автоматизированной проверки проверяет соответствие кода системы принятым стандартам кодирования и отсутствие в нем шаблонов часто встречающихся ошибок. Пользователь сам может добавить шаблон возможной ошибки или стандарта кодирования. Получаемые от инструмента сообщения анализируются на предмет серьезности обнаруженного дефекта и наличия ошибки в соответствующем коде. |

4. Отметьте те из перечисленных пунктов, которые правильно описывают тесты.

|  |  |
| --- | --- |
|  | Установить систему, после чего подготовить список обнаруженных во время процедуры установки проблем. |
|  | Запустить текстовый редактор и набрать 2-3 абзаца произвольного текста. Открыть диалог установки отступов. Задать нулевые отступы от левого и правого краев страницы и нажать кнопку OK. При этом должно появиться сообщение о некорректных значениях отступов. |
|  | Запустить текстовый редактор. В новом документе набрать фразу «Яр-Тур.~» (без кавычек). Сразу станет ясно, если это Microsoft Word 2000 без установленных обновлений. |
|  | Запустить калькулятор Windows. Набрать «4195835», затем «/», «3145727» и Enter. Если выдается ответ близкий к 1.33374, то процессор — Intel Pentium. На других типах процессоров будет ответ близкий к 1.33382. |

5. Поставьте в последнем столбце приведенной ниже таблицы числа, соответствующие покрытию данного кода с помощью указанных в первом столбце тестовых данных по критерию, указанному во втором.

При вычислении покрытия строк пустые строки и строки, содержащие только заголовок метода, do, { или } не учитываются. При вычислении покрытия условий текстуально одинаковые условия считаются как одно, если их значения всегда совпадают, и как разные, если при каком-то сценарии работы метода их значения могут оказаться различны.

 1 **int** gcd**(int** a, **int** b**)**

 2 **{**

 3 **if(**a == 0**)**

 4 **return** b;

 5 **if(**b == 0**)**

 6 **return** a;

 7 **if**(a == b)

 8 **return** a;

 9 **if(**a > 0 **&&** b < 0 **||** a < 0 **&&** b > 0**)**

10 b = -b;

11

12 **do**

13 **{**

14 **if(**b > a **&&** a > 0 **||** b < a **&&** a < 0**)**

15 **{**

16 a = b-a;

17 b = b-a;

18 a = a+b;

19 **}**

20

21 b = a-b;

22 a = a-b;

23 **}**

24 **while(**b **!**= 0**);**

25

26 **return** a;

27 **}**

|  |  |  |
| --- | --- | --- |
| (0, 3) | Покрывает из 16 строк |  |
| (0, 3) | Покрывает из 12 ветвлений |  |
| (0, 3) | Покрывает из 20 условий |  |
| (-2, 5) | Покрывает из 16 строк |  |
| (-2, 5) | Покрывает из 12 ветвлений |  |
| (-2, 5) | Покрывает из 20 условий |  |
| (0, 3), (-2, 5) | Покрывает из 16 строк |  |
| (0, 3), (-2, 5) | Покрывает из 12 ветвлений |  |
| (0, 3), (-2, 5) | Покрывает из 20 условий |  |

6. Отметьте в первом столбце таблицы те последовательности, которые являются различающими для изображенного автомата.

A/Y

A/X

B/X

B/Y

B/Y

A/Y

A/Y

B/X

|  |  |
| --- | --- |
|  | AAA |
|  | BAA |
|  | BBB |
|  | BBA |

По окончании учебного периода проводится письменный экзамен.

**Содержание возможных заданий на экзамене**

1. Критерии покрытия кода: покрытие ветвлений, условий, комбинаций условий, модифицированное покрытие условий и ветвлений (MC/DC). Составление полного по критерию MC/DC тестового набора для метода.
2. Доменное тестирование с линейными ограничениями. Выбор тестовых данных для метода с помощью техники доменного тестирования.
3. Покрывающие наборы и методы их построения. Использование покрывающих наборов при создании тестов на основе грамматик.
4. Нахождение диагностирующей последовательности и характеризующего множества для конечного автомата. Построение полного набора тестов для конечно-автоматной модели с помощью W-метода и Wp-метода.
5. **РЕСУРСЫ**
	1. **Основная литература**
	2. Материалы курса https://sites.google.com/site/swtestcourse/
6. R. Binder. Testing Object-Oriented Systems: Models, Patterns, and Tools. Addison-Wesley, 1999.
7. M. Broy, B. Jonsson, J.-P. Katoen, M. Leucker, A. Pretschner (eds.) Model Based Testing of Reactive Systems. LNCS 3472, Springer, 2005.
8. M. Utting, B. Legeard. Practical Model-Based Testing: A Tools Approach, M-K, 2006.
9. Д. Месарош. Шаблоны тестирования xUnit. М.: Вильямс, 2008.
10. C. Kaner, R. L. Fiedler. Foundations of Software Testing. Context-Driven Press, 2013.
	1. **Дополнительная литература**
11. B. Beizer. Software Testing Techniques. 2-nd edition. Int. Thomson Publishing, 1990.
12. B. Marick. The Craft of Software Testing, Prentice Hall, 1995.
13. В. В. Липаев. Тестирование программ. М., Радио и связь, 1986.
14. Г. Майерс. Искусство тестирования программ. М.: Финансы и статистика, 1982.
15. С. Канер, Дж. Фолк, Е. К. Нгуен. Тестирование программного обеспечения. М: Диасофт, 2000.
16. P. Bourque and R.E. Fairley, eds.,Guide to the Software Engineering Body of Knowledge, Version 3.0, IEEE Computer Society, 2014. https://www.computer.org/web/swebok/v3
17. А. Коберн. Современные методы описания требований к системам. М.: Лори, 2002.
18. В.В.Кулямин, Н.В.Пакулин, О.Л.Петренко, А.А.Сортов, А.В.Хорошилов. [Формализация требований на практике.](http://www.ispras.ru/publications/2006/formalizatsiya_trebovaniy_na_praktike/) Препринт Института системного программирования РАН, №13, 2006.
	1. **Программное обеспечение**

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Наименование** | **Условия доступа** |
| 1. |  Microsoft Windows 7 Professional RUSMicrosoft Windows 10Microsoft Windows 8.1 Professional RUS | *Из внутренней сети университета (договор)* |
| 2. | Microsoft Office Professional Plus 2010 | *Из внутренней сети университета (договор)* |
| 3. | Eclipse | *Свободно распространяемое ПО* |
| 4. | Mozilla Firefox | *Свободно распространяемое ПО* |
| 5. | AutoIt | *Свободно распространяемое ПО* |
| 6. | Selenium | *Свободно распространяемое ПО* |
| 7. | TestNG | *Свободно распространяемое ПО* |

* 1. **Профессиональные базы данных, информационные справочные системы,
	интернет-ресурсы (электронные образовательные ресурсы)**

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Наименование** | **Условия доступа** |
|  | ***Профессиональные базы данных, информационно-справочные системы*** |
| 1. | Электронно-библиотечная система Юрайт  | URL: https://biblio-online.ru/ |
|  | ***Интернет-ресурсы (электронные образовательные ресурсы)*** |
| 1. | Открытое образование  | URL: https://openedu.ru/ |

* 1. **Материально-техническое обеспечение дисциплины**

Учебные аудитории для лекционных занятий по дисциплине обеспечивают использование и демонстрацию тематических иллюстраций, соответствующих программе дисциплины в составе:

ПЭВМ с доступом в Интернет (операционная система, офисные программы, антивирусные программы);

мультимедийный проектор с дистанционным управлением.

Учебные аудитории для лабораторных и самостоятельных занятий по дисциплине оснащены компьютерами, с возможностью подключения к сети Интернет и доступом к электронной информационно-образовательной среде НИУ ВШЭ.