

Министерство науки и высшего образования Российской  
Федерации

Томский государственный университет  
систем управления и радиоэлектроники

А.А. Голубева

**ВЕРИФИКАЦИЯ ПРОГРАММНОГО  
ОБЕСПЕЧЕНИЯ**

Методические указания к практическим работам и организации  
самостоятельной работы для студентов направления  
«Программная инженерия»  
(уровень магистратуры)

Томск  
2021

УДК 004  
ББК 16  
Г621

**Рецензент:**

**Сидоров А. А.**, заведующий кафедрой автоматизации обработки информации Томского государственного университета систем управления и радиоэлектроники, канд. экон. наук, доцент

**Голубева Александра Александровна**

Верификация программного обеспечения: Методические указания к практическим работам и организации самостоятельной работы для студентов направления «Программная инженерия» (уровень магистратуры) / А.А. Голубева. – Томск, 2021. – 16 с.

Методические указания содержат описание практических занятий и рекомендации по организации самостоятельной работы студентов в рамках изучения дисциплины «Верификация программного обеспечения» и приобретения практических навыков направленных на изучение и закрепление на практике основных понятий, принципов и подходов верификации программного обеспечения.

Одобрено на заседании кафедры АОИ, протокол № 01 от 18.02.2021

УДК 004  
ББК 16  
Г621

© Голубева А.А., 2021  
© Томский государственный университет систем управления и радиоэлектроники, 2021

## Оглавление

1 ВВЕДЕНИЕ.....	4
2 МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ПРОВЕДЕНИЮ ПРАКТИЧЕСКИХ ЗАНЯТИЙ .....	4
2.1 Практическое занятие «Составление технического задания и проектной документации программного обеспечения» .....	5
2.2 Практическое занятие «Составление тест-кейсов. Применение методов верификации данных» .....	7
2.3 Практическое занятие «Оформление результатов тестирования» .....	8
3 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОРГАНИЗАЦИИ САМОСТОЯТЕЛЬНОЙ РАБОТЫ .....	11
3.1 Общие положения .....	11
3.2 Подготовка к контрольным работам .....	12
3.3 Подготовка к практическим занятиям .....	13
3.4 Проработка лекционного материала .....	13
3.5 Индивидуальное задание «Модели жизненного цикла программного обеспечения» .....	14
4 ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА .....	16

## **1 ВВЕДЕНИЕ**

Целью практических и самостоятельных работ в рамках изучения дисциплины «Верификация программного обеспечения» является формирование у студентов, обучающихся по направлению «Бизнесинформатика», навыков, позволяющих применять методы верификации программного обеспечения к различным типам проектов (задач).

Практические занятия направлены на изучение и закрепление на практике основных понятий, принципов и подходов верификации программного обеспечения.

Самостоятельная работа предполагает следующие виды деятельности: проработка лекционного материала для подготовки к практическим и самостоятельным работам, подготовка к контрольным работам, выполнение индивидуального задания.

## **2 МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ПРОВЕДЕНИЮ ПРАКТИЧЕСКИХ ЗАНЯТИЙ**

### **2.1 Практическое занятие «Составление технического задания»**

#### **Цель занятия**

Целью проведения практического задания является приобретение практических навыков и теоретических знаний по разработке технического задания.

#### **Рекомендации по подготовке к занятию**

Во время подготовки к практическому заданию рекомендуется ознакомиться с лекционным материалом и просмотреть дополнительную учебно-методическую литературу:

1. ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению [Электронный ресурс]. URL: [https://www.ptj-exr.ru/gost/gost\\_19-201-78.php](https://www.ptj-exr.ru/gost/gost_19-201-78.php) (дата обращения 10.07.2018)

#### **Содержание занятия / Порядок проведения занятия**

1. Выбрать предметную область (проект, продукт, решение) к которому будет разрабатываться техническое задание и проектная документация.

2. Определить основные функции и ограничения системы.

3. Разработать техническое задание к проекту.

4. Промоделировать процессы обсуждения, корректировки и утверждения технического задания с заказчиком.

5. Подготовить 2-3 минутный доклад, в котором отразить основные особенности процесса: чем пользовались, с чем столкнулись, основные сложности, как решили.

Во время проведения практического занятия каждый студент выбирает предметную область (задачу) в рамках которой он будет прорабатывать техническое задание к разрабатываемому проекту.

## **2.2 Практическое занятие «Составление проектной документации программного обеспечения»**

### **Цель занятия**

Целью проведения практического задания является приобретение практических навыков и теоретических знаний по проектной документации к проекту.

### **Рекомендации по подготовке к занятию**

Во время подготовки к практическому заданию рекомендуется ознакомиться с лекционным материалом и просмотреть дополнительную учебно-методическую литературу:

1. Золотов, С. Ю. Проектирование информационных систем: Учебное пособие [Электронный ресурс] / Золотов С. Ю. — Томск: ТУСУР, 2016. — 117 с. — Режим доступа: <https://edu.tusur.ru/publications/6478>.

### **Содержание занятия / Порядок проведения занятия**

1. Выбрать предметную область (проект, продукт, решение) к которому будет разрабатываться проектная документация.
2. Определить основные функции и ограничения системы.
3. Разработать проектную документацию к проекту.
4. Промоделировать процессы обсуждения, корректировки и утверждения т проектной документации с заказчиком.
5. Подготовить 2-3 минутный доклад, в котором отразить основные особенности процесса: чем пользовались, с чем столкнулись, основные сложности, как решили.

Во время проведения практического занятия каждый студент выбирает предметную область (задачу) в рамках которой он будет прорабатывать проектную документацию к разрабатываемому проекту.

### **2.3 Практическое занятие «Составление тест-кейсов»**

#### **Цель занятия**

Целью проведения практического задания является приобретение практических навыков и расширение знаний по построению тест-кейсов системы.

#### **Рекомендации по подготовке к занятию**

Во время подготовки к практическому заданию рекомендуется ознакомиться с лекционным материалом и просмотреть дополнительную учебно-методическую литературу:

1. Верификация программного обеспечения [Электронный ресурс]. URL: [http://window.edu.ru/resource/700/41700/files/verif\\_po.pdf](http://window.edu.ru/resource/700/41700/files/verif_po.pdf) (дата обращения 10.07.2018)

#### **Содержание занятия / Порядок проведения занятия**

1. Выбрать предметную область (проект, продукт, решение) к которому будет разрабатываться техническое задание и проектная документация.

2. Выбрать 1-3 наиболее подходящие к проекту (продукту, решению) методы верификации программного обеспечения.

3. На основе отобранных методов верификации данных разработать тест-кейсы к системе.

4. Подготовить 2-3 минутный доклад, в котором отразить основные особенности процесса: чем пользовались, с чем столкнулись, основные сложности, как решили.

Во время проведения практического занятия каждый студент продолжает прорабатывать выбранную на практическом занятии «Составление технического задания и проектной документации программного обеспечения» предметную область.

## **2.4 Практическое занятие «Применение методов верификации данных»**

### **Цель занятия**

Целью проведения практического задания является приобретение практических навыков и расширение знаний по применению методов верификации к оценке качества программного продукта.

### **Рекомендации по подготовке к занятию**

Во время подготовки к практическому заданию рекомендуется ознакомиться с лекционным материалом и просмотреть дополнительную учебно-методическую литературу:

1. Верификация программного обеспечения [Электронный ресурс]. URL: [http://window.edu.ru/resource/700/41700/files/verif\\_po.pdf](http://window.edu.ru/resource/700/41700/files/verif_po.pdf) (дата обращения 10.07.2018)

### **Содержание занятия / Порядок проведения занятия**

1. Выбрать предметную область (проект, продукт, решение) к которому будет разрабатываться техническое задание и проектная документация.

2. Выбрать 1-3 наиболее подходящие к проекту (продукту, решению) методы верификации программного обеспечения.

3. Провести процесс верификации программного обеспечения по выбранным в п.3 методам.

4. Промоделировать процесс взаимодействия тестировщика и разработчика.

5. Подготовить 2-3 минутный доклад, в котором отразить основные особенности процесса: чем пользовались, с чем столкнулись, основные сложности, как решили.

### **Варианты заданий**

Во время проведения практического занятия каждый студент продолжает прорабатывать выбранную на практическом занятии «Составление технического задания и проектной документации программного обеспечения» предметную область.

## **2.5 Практическое занятие «Оформление результатов тестирования»**

### **Цель занятия**

Целью проведения практического задания является приобретение практических навыков и расширение знаний по оформлению результатов тестирования; ставлению тест-кейсов.

### **Рекомендации по подготовке к занятию**

Во время подготовки к практическому заданию рекомендуется ознакомиться с лекционным материалом и просмотреть дополнительную учебно-методическую литературу, например, Методы верификации программного обеспечения [Электронный ресурс].

URL: <http://www.ict.edu.ru/ft/005645/62322e1-st09.pdf> (дата обращения 10.07.2018)

### **Содержание занятия / Порядок проведения занятия**

1. Выбрать инструмент фиксирования результатов применения методов верификации программного обеспечения.

2. Зафиксировать в нем результаты, полученные при выполнении практического занятия «Составление тест-кейсов. Применение методов верификации данных».

3. Промоделировать процесс взаимодействия с разработчиком по исправлению багов системы.

4. Подготовить 2-3 минутный доклад, в котором отразить основные особенности процесса: чем пользовались, с чем столкнулись, основные сложности, как решили.

### **Варианты заданий**

Во время проведения практического занятия каждый студент продолжает прорабатывать выбранную на практическом занятии «Составление технического задания и проектной документации программного обеспечения» предметную область.

## **3 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОРГАНИЗАЦИИ САМОСТОЯТЕЛЬНОЙ РАБОТЫ**

### **3.1 Общие положения**

Самостоятельная работа студентов рассматривается как вид деятельности, позволяющий целенаправленно формировать и развивать самостоятельность студента как личностное качество при выполнении различных видов заданий и проработке дополнительного учебного материала.

Целями самостоятельной работы являются систематизация, расширение и закрепление теоретических знаний об основах верификации программного обеспечения.

Самостоятельная работа студента включает следующие виды его активности:

1. Подготовка к контрольным работам.
2. Выполнение индивидуальных заданий.

3. Проработка лекционного материала.

4. Подготовка к практическим занятиям.

Критериями оценки внеаудиторной самостоятельной работы студентов могут быть:

- уровень развития логического мышления студента (гибкость, рациональность, оригинальность мышления);
- сформированность умений самообразования студента (способность находить, систематизировать и применять информацию из различных источников для решения поставленных задач);
- степень развития коммуникативных умений (умение работать в малых группах, выступать с докладом);
- грамотность в изложении материала;
- сформированность самоконтроля и самооценки.

Самостоятельная работа является важной составляющей в изучении дисциплины и заключается в самостоятельном изучении теоретического материала, подготовки к выполнению контрольных работ, подготовки к практическим занятиям.

### **3.2 Подготовка к контрольным работам**

В процессе изучения дисциплины «Верификация программного обеспечения» студент должен выполнить ряд контрольных работ, главная цель которых – оказать студенту помощь в его работе. Оценки и замечания к выполненным работам позволяют студенту судить о степени усвоения им соответствующего раздела курса; указывают на имеющиеся у него пробелы, на желательное направление дальнейшей работы; помогают сформулировать вопросы для постановки их перед преподавателем.

Не следует приступать к выполнению контрольного задания, не решив достаточного количества задач по материалу, соответствующему этому заданию. Опыт показывает, что чаще всего неумение решить ту или иную задачу контрольного задания вызывается тем, что студент не выполнил это требование.

Для успешных выполнений контрольных работ рекомендуется детально проработать лекционный материал по темам:

1. Основные понятия и определения в процессе верификации программного обеспечения.
2. Роль верификации в жизненном цикле разработки программного обеспечения.
3. Модели жизненного цикла разработки программного обеспечения.
4. Задачи и цели процесса верификации.
5. Современные технологии разработки программного обеспечения.
6. Тестирование программного обеспечения.
7. Методологии управления проектами.

Контрольные работы должны выполняться самостоятельно. Несамостоятельно выполненная работа не дает возможности преподавателю указать студенту на недостатки в его работе, в усвоении им учебного материала, в результате чего студент не приобретает необходимых знаний и может оказаться неподготовленным к экзамену (зачету).

### **3.3 Подготовка к практическим занятиям**

Практические задания предназначены для верификации полученных знаний и закрепления теоретической части дисциплины.

На практических занятиях подробно рассматриваются основные вопросы дисциплины, разбираются основные типы задач. К практическому занятию следует заранее повторить лекционный материал к следующей теме. Систематическое выполнение домашних заданий обязательно и является важным фактором, способствующим успешному усвоению дисциплины.

Для подготовки к практическим занятиям рекомендуется детально проработать лекционный материал по темам:

1. Роль верификации в жизненном цикле разработки программного обеспечения.
2. Модели жизненного цикла разработки программного обеспечения.
3. Задачи и цели процесса верификации. Методы верификации программного обеспечения.

Во время проработки лекционного материала необходимо отработать прослушанную лекцию (прочитать конспект, познакомиться с дополнительной литературой) и восполнить пробелы в знаниях, если таковые обнаружались.

На каждом практическом занятии проводится либо устный, либо тестовый опрос по пройденным темам, которые позволят определить уровень подготовки и степень готовности к выполнению контрольной работы по данной дисциплине

### **3.4 Проработка лекционного материала**

В рамках проработки лекционного материала необходимо:

- 1) отработать прослушанную лекцию (прочитать конспект, просмотреть презентационный материал) и восполнить пробелы в знаниях, если таковые обнаружались;
- 2) перед каждой последующей лекцией повторно прочитать конспект по предыдущей, чтобы обновить знания для восприятия последующей – новой – информации.

Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного понимания предыдущего, производя на бумаге все вычисления (в том числе и те, которые ради краткости опущены в учебнике) и выполняя имеющиеся в учебнике задания для самопроверки. Особое внимание следует обращать на определение основных понятий.

Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. При изучении материала по учебнику полезно вести конспект, в который рекомендуется вписывать определения, формулировки теорем, формулы, уравнения и т. д. На полях конспекта следует отмечать вопросы, выделенные студентом для получения письменной или устной консультации преподавателя.

При решении задач нужно обосновать каждый этап решения исходя из теоретических положений курса. Если студент видит несколько путей решения, то он должен сравнить их и выбрать из них самый лучший. Полезно до начала вычислений составить краткий план решения.

### **3.5 Индивидуальное задание «Модели жизненного цикла программного обеспечения»**

#### **Цель индивидуального задания**

Целью индивидуального задания является систематизация, расширение и закрепление теоретических знаний, по моделям жизненного цикла программного обеспечения.

#### **Порядок выполнения и содержание работ**

1. Получить вариант задания у преподавателя.
2. Проработать теоретические основы предметной области.
3. Показать применение модели жизненного цикла на практической задаче.
4. Определить достоинства и недостаток модели жизненного цикла.
5. Подготовить 5-и минутный доклад, в котором отразить ход работы и результаты.

## **Примеры индивидуальных заданий Модели**

жизненного цикла:

1. Спиральная модель.
2. V-образная модель.
3. Каскадная модель.

Доклад – это самостоятельная работа, анализирующая и обобщающая публикации по заданной тематике, предполагающая выработку и обоснование собственной позиции докладчика в отношении рассматриваемых вопросов. Основным содержанием доклада может быть описание состояния дел в какой-либо научной или практической сфере; авторский взгляд на ситуацию или проблему, анализ и возможные пути решения проблемы.

При подготовке к докладу необходимо подготовить план выступления, выстроить доклад в краткой лаконичной форме, последовательно, с соблюдением логических связей между фрагментами выступления. Речь должна быть грамотной и выжатой. Желательно по время выступления не читать весь текст. Следует продумать взаимосвязь выступления с показом демонстрационных материалов.

Работа студента над докладом-презентацией включает отработку умения самостоятельно обобщать материал и делать выводы в заключении, умения ориентироваться в материале и отвечать на дополнительные вопросы слушателей, отработку навыков ораторства, умения проводить диспут.

Докладчики должны знать и уметь: сообщать новую информацию; использовать технические средства; хорошо ориентироваться в теме всего семинарского занятия; дискутировать и быстро отвечать на заданные вопросы; четко выполнять установленный регламент (не более 5 минут); иметь представление о композиционной структуре доклада.

## 5 СПИСОК ЛИТЕРАТУРЫ

1. Казарин О.В. Надежность и безопасность программного обеспечения [Электронный ресурс] : учеб. пособие / О.В. Казарин, И.Б. Шубинский. – М.: Издательство Юрайт, 2018. – 342 с. – Режим доступа: <https://biblio-online.ru/book/6A637EC7-8B784DA6-B404-71DE0202E2EF> (дата обращения:10.07.2018).

2. Плаксин М.А. Тестирование и отладка программ для профессионалов будущих и настоящих [Электронный ресурс]. – М. : БИНОМ. Лаборатория знаний, 2015. – 170 с. – Режим доступа : <https://ibooks.ru/reading.php?productid=353395> (дата обращения: 10.07.2018).

3. Калайда, В. Т. Теория вычислительных процессов: Методическое пособие [Электронный ресурс] / В. Т. Калайда. — Томск: ТУСУР, 2012. — 135 с. — Режим доступа: <https://edu.tusur.ru/publications/2048> (дата обращения:10.07.2018)