

УТВЕРЖДАЮ
 Декан факультета

 (подпись) Матвеев П.В.
 ФИО
 «___» _____ 20__

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ ТЕОРИЯ ВЕРИФИКАЦИИ И ВАЛИДАЦИИ ПАРАЛЛЕЛЬНЫХ И РАСПРЕДЕЛЕННЫХ ПРОГРАММНЫХ СИСТЕМ

| | |
|--|--|
| Направление/специальность подготовки | 09.04.04 Программная инженерия |
| Специализация/профиль/программа подготовки | Процессы и методы разработки программных продуктов |
| Уровень высшего образования | Магистратура |
| Форма обучения | Заочная |
| Факультет | О Естественнонаучный |
| Выпускающая кафедра | О7 Информационные системы и программная инженерия |
| Кафедра-разработчик рабочей программы | О7 Информационные системы и программная инженерия |

| КУРС | СЕМЕСТР | ОБЩАЯ ТРУДОЁМКОСТЬ (ЗАЧЕТНЫХ ЕДИНИЦ) | ЧАСЫ (по наличию видов занятий) | | | | | | | | | ВИД ПРОМЕЖУТОЧНОГО КОНТРОЛЯ |
|------|---------|---|---------------------------------|--------------------|--------|---------------------------|-------------------------|------------------------|-----------------|-----------------|-------------------------------|--------------------------------|
| | | | ОБЩАЯ ТРУДОЁМКОСТЬ | АУДИТОРНЫЕ ЗАНЯТИЯ | | | | САМОСТОЯТЕЛЬНАЯ РАБОТА | | | | |
| | | | | ВСЕГО | ЛЕКЦИИ | ЛАБОРАТОРНЫЙ ПРАКТИКУМ | ПРАКТИЧЕСКИЕ ЗАНЯТИЯ | ВСЕГО | КУРСОВОЙ ПРОЕКТ | КУРСОВАЯ РАБОТА | ДРУГИЕ ВИДЫ САМОСТ. РАБОТЫ | |
| 2 | 4 | 3 | 108 | 8 | 4 | 0 | 4 | 100 | 0 | 0 | 100 | зач. |

ЛИСТ СОГЛАСОВАНИЯ

РАБОЧАЯ ПРОГРАММА СОСТАВЛЕНА В СООТВЕТСТВИИ С ТРЕБОВАНИЯМИ ФЕДЕРАЛЬНОГО
ГОСУДАРСТВЕННОГО ОБРАЗОВАТЕЛЬНОГО СТАНДАРТА ВЫСШЕГО ОБРАЗОВАНИЯ (ФГОС ВО)

09.04.04 Программная инженерия

год набора группы: 2024

Программу составили:

Кафедра О7 Информационные системы и программная инженерия
Смирнова Мария Сергеевна, д.т.н., доцент, профессор

Кафедра О7 Информационные системы и программная инженерия
Садырова Айганыш Кылычбековна, преподаватель

Программа рассмотрена
на заседании кафедры-разработчика
рабочей программы **О7 Информационные системы и программная инженерия**

Заведующий кафедрой Семенова Е.Г., д.т.н., проф.

Программа рассмотрена
на заседании выпускающей кафедры

О7 Информационные системы и программная инженерия

Заведующий кафедрой Семенова Е.Г., д.т.н., проф.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
ТЕОРИЯ ВЕРИФИКАЦИИ И ВАЛИДАЦИИ ПАРАЛЛЕЛЬНЫХ И РАСПРЕДЕЛЕННЫХ
ПРОГРАММНЫХ СИСТЕМ**

Разделы рабочей программы

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО
3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ
4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ
5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ
6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Приложения к рабочей программе дисциплины

- Приложение 1. Аннотация рабочей программы
- Приложение 2. Технологии и формы обучения
- Приложение 3. Фонды оценочных средств

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является формирование следующих компетенций:

| |
|--|
| ПСК-2.2 — способность обеспечить управление архитектурой интегрированного программного обеспечения и единой информационной среды |
| ОПК-4 — способность применять на практике новые научные принципы и методы исследований |

Формированию компетенций служит достижение следующих результатов образования:

ПСК-2.2

знания:

На уровне представлений: место верификации и валидации в жизненном цикле программных систем; основной математический аппарат, применяемый при верификации и валидации программных систем; характеристики качества параллельных и распределенных программных систем.

На уровне воспроизведения: классификация методов верификации и валидации при разработке и сопровождении программных систем; понятие формальных методов верификации программных систем;

умения:

использовать основные подходы к автоматизации процессов верификации программных систем.;

ОПК-4

знания:

На уровне представлений: место верификации и валидации в жизненном цикле программных систем; основной математический аппарат, применяемый при верификации и валидации программных систем; характеристики качества параллельных и распределенных программных систем.

На уровне воспроизведения: классификация методов верификации и валидации при разработке и сопровождении программных систем; понятие формальных методов верификации программных систем; понятие сильной и слабой бисимуляции агентов; понятие статических, динамических и синтетических методов верификации программных систем.

На уровне понимания: логико-алгебраические, исполнимые и промежуточные модели формальные модели требований, поведения и окружения программного обеспечения; общие и специализированные экспертные методы верификации и валидации в жизненном цикле программных систем; тестирование как динамический метод верификации программных систем;;

умения:

Теоретические: применять общие принципы организации верификации и валидации программных систем; использовать основные подходы к автоматизации процессов верификации программных систем.

Практические: задавать требования к характеристикам качества параллельных и распределенных программных систем с учетом специфики предметной области; владеть методами определения эквивалентности и конгруэнции агентов;;

навыки:

Построения формальных спецификаций программных систем; построения формальных моделей программных систем; разработки тестов.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО

Дисциплина **ТЕОРИЯ ВЕРИФИКАЦИИ И ВАЛИДАЦИИ ПАРАЛЛЕЛЬНЫХ И РАСПРЕДЕЛЕННЫХ ПРОГРАММНЫХ СИСТЕМ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению *09.04.04 Программная инженерия*.

Содержание дисциплины является логическим продолжением дисциплин: **МЕТОДОЛОГИЯ ПРОГРАММНОЙ ИНЖЕНЕРИИ**.

Содержание дисциплины является основой для освоения дисциплин: **ВЫПОЛНЕНИЕ И ЗАЩИТА ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ**.

Предварительные компетенции, сформированные у обучающегося до начала изучения дисциплины:

- ОПК-1 — Способен самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте
- ПСК-2.1 — Способен выполнить постановку новых задач анализа и синтеза новых проектных решений

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 3 з.е., 108 ч.

3.1. Содержание (дидактика) дисциплины

| КУРС | СЕМЕСТР | Наименование разделов и дидактических единиц | ВСЕГО | Аудиторные занятия в контактной форме | | | Самостоятельная работа студентов | Формируемая компетенция, % | |
|---------------------|---------|--|-------|---------------------------------------|--------|----------------------|----------------------------------|----------------------------|-------|
| | | | | ВСЕГО | Лекции | Практические занятия | | ПСК-2.2 | ОПК-4 |
| | | | | | | | | | |
| 2 | 4 | Раздел 1. Верификация и валидация как часть жизненного цикла программных систем. 1.1. Виды деятельности, роли и артефакты в жизненном цикле программных систем. 1.2. Верификация и валидация как виды деятельности при разработке и сопровождении программных систем. 1.3. Характеристики качества параллельных и распределенных программных систем. 1.4. Классификация методов верификации и валидации при разработке и сопровождении программных систем. | 33 | 2 | 1 | 1 | 31 | 25 | 25 |
| 2 | 4 | Раздел 2. Формальные методы верификации программного обеспечения. 2.1. Классификация формальных методов верификации программного обеспечения. 2.2. Алгебры процессов. 2.3. Исчисление взаимодействующих систем. Именованные системы переходов. 2.4. Эквивалентность поведения. Конгруэнции. 2.5. Эквивалентность множества путей. 2.6. Бисимуляция и наблюдаемая эквивалентность. 2.7. Алгоритмы вычисления бисимуляции. 2.8. Сети Петри, логики Хоара, программные контракты. 2.9. Метод верификации на модели. | 16 | 2 | 1 | 1 | 14 | 25 | 25 |
| 2 | 4 | Раздел 3. Экспертные методы верификации и валидации в жизненном цикле программных систем. 3.1. Общие экспертные методы верификации и валидации. 3.2. Специализированные экспертные методы: организационная экспертиза, эвристическая оценка пользовательского интерфейса, аудит защищенности программных систем. 3.3. Методы анализа архитектуры программного обеспечения. | 27 | 2 | 1 | 1 | 25 | 25 | 25 |
| 2 | 4 | Раздел 4. Современные и перспективные подходы к автоматизации процессов верификации программных систем. 6.1. Автоматизация процессов верификации формальными методами. 6.2. Автоматизация статической верификации артефактов жизненного цикла программных систем. 6.3. Автоматизация тестирования программного обеспечения. 6.4. Подходы к автоматизации синтетических методов верификации. | 32 | 2 | 1 | 1 | 30 | 25 | 25 |
| Всего за 4 семестр | | | 108 | 8 | 4 | 4 | 100 | 100 | 100 |
| Всего по дисциплине | | | 108 | 8 | 4 | 4 | 100 | 100 | 100 |

3.2. Аудиторный практикум

| № п/п | Номер и наименование раздела дисциплины | Тема практического занятия | Объем, ауд. часов |
|---------------------------|---|--|-------------------|
| 1 | Раздел 1. Верификация и валидация как часть жизненного цикла программных систем. | Верификация и валидация артефактов жизненного цикла программных систем | 0.5 |
| 2 | | Описание характеристик качества программного обеспечения | 0.5 |
| 3 | Раздел 2. Формальные методы верификации программного обеспечения. | Верификация поведения программных агентов формальными методами (контрольная работа №1) | 0.4 |
| 4 | | Описание процессов в исчислении взаимодействующих систем. Спецификация протокола как системы взаимодействующих агентов | 0.2 |
| 5 | | Вычисление бисимуляции и определение вида эквивалентности программных агентов | 0.4 |
| 6 | Раздел 3. Экспертные методы верификации и валидации в жизненном цикле программных систем. | Общие и специализированные экспертные методы верификации и валидации в жизненном цикле программных систем. Систематические методы анализа архитектуры программного обеспечения | 1 |
| 7 | Раздел 4. Современные и перспективные подходы к автоматизации процессов верификации программных систем. | Автоматизация процессов верификации программных систем | 1 |
| Всего за 4 семестр | | | 4 |

3.3. Самостоятельная работа студента (СРС)

| № п/п | Номер и наименование раздела дисциплины | Содержание учебного задания | Объем, часов |
|--------------------|---|--|-----------------|
| 1 | Раздел 1. Верификация и валидация как часть жизненного цикла программных систем. | Изучение предусмотренных программой материалов по рекомендуемой литературе | 16 |
| 2 | | Подготовка к практическим занятиям | 15 |
| 3 | Раздел 2. Формальные методы верификации программного обеспечения. | Подготовка к практическим занятиям | 6 |
| 4 | | Подготовка к контрольной работе | 2 |
| 5 | | Изучение предусмотренных программой материалов по рекомендуемой литературе | 4 |
| 6 | | Выполнение домашнего задания | 2 |
| 7 | Раздел 3. Экспертные методы верификации и валидации в жизненном цикле программных систем. | Изучение предусмотренных программой материалов по рекомендуемой литературе | 15 |
| 8 | | Подготовка к практическим занятиям | 10 |
| 9 | Раздел 4. Современные и перспективные подходы к автоматизации процессов верификации программных систем. | Изучение предусмотренных программой материалов по рекомендуемой литературе | 15 |
| 10 | | Подготовка к практическим занятиям | 15 |
| Всего за 4 семестр | | | 100 |

4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

| СЕМЕСТР | НЕДЕЛИ СЕМЕСТРА | | | | | | | | | | | | | | | | |
|----------|-----------------|---|---|---|---|----|---|---|---|----|----|----|----|----|----|----|------|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
| 4 | | | | | | ДР | | | | ДР | | | | | ДЗ | ДР | зач. |

Условные обозначения:

- ДР – диагностическая работа;
- ДЗ – домашнее задание;
- зач. – зачет.

Текущий контроль успеваемости студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- домашнее задание.

Промежуточная аттестация проводится в формах:

- зачет.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Основная литература по дисциплине:

1. . Управление программными проектами. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2013, эл. рес.
2. И. С. Петухов. . Разработка программного обеспечения. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007, эл. рес.
3. Н. Н. Смирнова. . Верификация и тестирование программных систем. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014, эл. рес.
4. Н. Н. Смирнова, С. Д. Тарасов. . Основы построения компиляторов. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007, эл. рес.
5. С. В. Сеницын, Н. Ю. Налютин. . Верификация программного обеспечения. М.: Интернет-Университет Информационных Технологий, 2007, эл. рес.
6. С. М. Старолетов. . Основы тестирования и верификации программного обеспечения. Санкт-Петербург: Лань, 2020, эл. рес.

5.2. Дополнительная литература по дисциплине:

не требуется.

5.3. Периодические издания:

не требуются.

5.4. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины, электронные библиотечные системы:

1. http://library.voenmeh.ru/jirbis2/index.php?option=com_irbis&view=irbis&Itemid=474 — Фундаментальная библиотека БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова;
2. <https://e.lanbook.com/> — ЭБС Лань;
3. <https://ibooks.ru/> — ЭБС Айбукс.ру - это большой выбор актуальной литературы для вашей библиотеки в электронном виде;
4. <https://urait.ru/> — Образовательная платформа «Юрайт». Для вузов и ссузов.;
5. <http://www.emis.de/ELibM.html> - Electronic Library of Mathematics.;
6. <https://spinroot.com/spin/whatispin.html> Spin - Formal Verification;
7. <https://frama-c.com/> — Frama-C - Framework for Modular Analysis of C programs;
8. <https://robotframework.org/> — Robot Framework.

Современные профессиональные базы данных:

1. <https://rusneb.ru> – Национальная электронная библиотека (НЭБ);
2. <https://cyberleninka.ru/> - Научная электронная библиотека «Киберленинка»;
- <http://www.rfbr.ru/rffi/ru/library> - Полнотекстовая электронная библиотека Российского фонда фундаментальных исследований.

Информационные справочные системы:

1. Техэксперт – Информационный портал технического регулирования: Нормы, правила, стандарты РФ;
2. http://library.voenmeh.ru/jirbis2/index.php?option=com_irbis&view=irbis&Itemid=457 - БД ГОСТов собственной генерации БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова;
3. <http://www.consultant.ru/>- КонсультантПлюс- информационный портал правовой информации.

5.5. Программное обеспечение:

1. Kubuntu 18.04 LTS;
2. Офисный пакет Libre Office;
3. Интернет-браузер Mozilla Firefox;
4. Интернет-браузер Chromium;
5. Набор средств верификации Robot Framework;
6. Набор средств верификации Frama-C;

7. Набор средств верификации Spin.

5.6. Информационные технологии:

взаимодействие с обучающимися посредством ЭИОС Moodle БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Лекционные занятия:

специализированные требования по оборудованию отсутствуют; аудитория с посадочными местами по количеству студентов; доска.

6.2. Практические занятия:

1. Kubuntu 18.04 LTS;
2. Офисный пакет Libre Office;
3. Интернет-браузер Mozilla Firefox;
4. Интернет-браузер Chromium;
5. Набор средств верификации Robot Framework;
6. Набор средств верификации Frama-C;
7. Набор средств верификации Spin.

6.3. Прочее:

1. рабочее место преподавателя, оснащенное компьютером с доступом в Интернет;
2. рабочие места студентов, оснащенные компьютерами с доступом в Интернет, предназначенные для работы в электронной образовательной среде.

Аннотация рабочей программы

Дисциплина **ТЕОРИЯ ВЕРИФИКАЦИИ И ВАЛИДАЦИИ ПАРАЛЛЕЛЬНЫХ И РАСПРЕДЕЛЕННЫХ ПРОГРАММНЫХ СИСТЕМ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению *09.04.04 Программная инженерия*. Дисциплина реализуется на факультете О Естественнoнаучный БГТУ "ВОЕНМЕХ" им. Д.Ф. Устинова кафедрой О7 Информационные системы и программная инженерия.

Дисциплина нацелена на формирование *компетенций*:

ПСК-2.2 способность обеспечить управление архитектурой интегрированного программного обеспечения и единой информационной среды;

ОПК-4 способность применять на практике новые научные принципы и методы исследований.

Содержание дисциплины охватывает круг вопросов, связанных с основными понятиями, принципами, методами верификации и валидации программных систем. Основное внимание уделяется параллельным и распределенным программным системам, а также формальным методам верификации программных систем.

Программой дисциплины предусмотрены следующие **виды контроля**:

Текущий контроль успеваемости студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- домашнее задание.

Промежуточная аттестация проводится в формах:

- зачет.

Общая трудоемкость освоения дисциплины составляет 3 з.е., **108 ч**. Программой дисциплины предусмотрены лекционные занятия (**4 ч.**), практические занятия (**4 ч.**), самостоятельная работа студента (**100 ч**).

ТЕХНОЛОГИИ И ФОРМЫ ОБУЧЕНИЯ

Рекомендации по освоению дисциплины для студента

Трудоемкость освоения дисциплины составляет 108 ч., из них 8 ч. аудиторных занятий, и 100 ч., отведенных на самостоятельную работу студента.

Рекомендации по распределению учебного времени по видам самостоятельной работы и разделам дисциплины приведены в таблице.

Контроль освоения дисциплины производится в соответствии с Положением о текущем, рубежном контроле успеваемости и промежуточной аттестации обучающихся.

Формы контроля и критерии оценивания приведены в приложении 3 к Рабочей программе.

| Наименование работы | Рекомендуемая литература | Трудоемкость, час. |
|--|---|--------------------|
| Раздел 1. Верификация и валидация как часть жизненного цикла программных систем. | | |
| Изучение предусмотренных программой материалов по рекомендуемой литературе | С. В. Синицын, Н. Ю. Налютин. . Верификация программного обеспечения: М.: Интернет-Университет Информационных Технологий, 2007 (1.1-1.3, 1.6-1.8) С. М. Старолетов. . Основы тестирования и верификации программного обеспечения: Санкт-Петербург: Лань, 2020 (1.1) | 16 |
| Подготовка к практическим занятиям | И. С. Петухов. . Разработка программного обеспечения: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007 (1.2) Н. Н. Смирнова. . Верификация и тестирование программных систем: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1.1) | 15 |
| Итого по разделу 1 | | 31 |
| Раздел 2. Формальные методы верификации программного обеспечения. | | |
| Подготовка к практическим занятиям | Н. Н. Смирнова. . Верификация и тестирование программных систем: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1.1, 1.3-1.11) | 6 |
| Подготовка к контрольной работе | С. М. Старолетов. . Основы тестирования и верификации программного обеспечения: Санкт-Петербург: Лань, 2020 (2.1) | 2 |
| Изучение предусмотренных программой материалов по рекомендуемой литературе | И. С. Петухов. . Разработка программного обеспечения: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007 (1.2) С. В. Синицын, Н. Ю. Налютин. . Верификация программного обеспечения: М.: Интернет-Университет Информационных Технологий, 2007 (1.1-1.3, 1.6-1.8) | 4 |
| Выполнение домашнего задания | | 2 |
| Итого по разделу 2 | | 14 |
| Раздел 3. Экспертные методы верификации и валидации в жизненном цикле программных систем. | | |
| Изучение предусмотренных программой материалов по рекомендуемой литературе | И. С. Петухов. . Разработка программного обеспечения: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007 (1.1) Н. Н. Смирнова. . Верификация и тестирование программных систем: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1.1) . Управление программными проектами: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2013 (Теоретические сведения к лабораторным работам №1 и №3) | 15 |
| Подготовка к практическим занятиям | С. В. Синицын, Н. Ю. Налютин. . Верификация программного обеспечения: М.: Интернет-Университет Информационных Технологий, 2007 (1.5, 1.8-1.10, 6.1, 6.2, 9, 10) | 10 |
| Итого по разделу 3 | | 25 |

| Раздел 4. Современные и перспективные подходы к автоматизации процессов верификации программных систем. | | |
|---|--|----|
| Изучение предусмотренных программой материалов по рекомендуемой литературе | Н. Н. Смирнова. . Верификация и тестирование программных систем: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1.1-1.4, 1.9, 2.1-2.3, 2.6-2.8) С. В. Сеницын, Н. Ю. Налютин. . Верификация программного обеспечения: М.: Интернет-Университет Информационных Технологий, 2007 (1.9, 2-8, 11, 12) Н. Н. Смирнова, С. Д. Тарасов. . Основы построения компиляторов: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007 (2-4) | 15 |
| Подготовка к практическим занятиям | . Управление программными проектами: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2013 (Теоретические сведения к лабораторным работам №3 и №4) | 15 |
| Итого по разделу 4 | | 30 |

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств, позволяющие оценить результаты обучения по данной дисциплине, включают в себя:

- диагностическая работа
- домашнее задание;
- зачет.

Критерии оценивания

Диагностическая работа

Диагностическая работа проводится в форме теста в ЭИОС Moodle:

- при правильном ответе менее чем на 60% вопросов - не аттестация;
- при правильном ответе на 60% вопросов и более - аттестация.

Домашнее задание

Домашнее задание содержит набор задач на построение формальной спецификации и модели программной системы, получение описаний соответствующих агентов в исчислении взаимодействующих систем, построение именованных систем переходов, вычисление бисимуляции и определение вида эквивалентности агентов в соответствии с индивидуальным вариантом. Варианты индивидуальных заданий, требования к оформлению по каждому пункту задания приведены в УМК дисциплины.

Для оценки "удовлетворительно" необходимо правильно построить хотя бы один из процессных графов параллельной композиции трех агентов.

Для оценки "хорошо" необходимо правильно вычислить бисимуляцию хотя бы одной указанной в задании пары агентов, представленных соответствующими построенными процессными графами.

Для оценки "отлично" необходимо выполнить все пункты задания -- построить все процессные графы и вычислить бисимуляцию всех указанных пар агентов.

Зачет

Зачет в форме теста по вопросам, размещенным в Moodle. Тест содержит 10 вопросов, которые оцениваются суммарно в 10 баллов. Для получения зачета необходимо набрать не менее 6 баллов.

Паспорт фонда оценочных средств

| КУРС | СЕМЕСТР | Наименование разделов и дидактических единиц | ВСЕГО | Аудиторные занятия в контактной форме | | | Самостоятельная работа студентов | Формируемая компетенция, % | | НАИМЕНОВАНИЕ ОЦЕНОЧНОГО СРЕДСТВА |
|---------------------|---------|---|-------|---------------------------------------|--------|----------------------|----------------------------------|----------------------------|-------|----------------------------------|
| | | | | ВСЕГО | Лекции | Практические занятия | | ПСК-2.2 | ОПК-4 | |
| | | | | | | | | | | |
| 2 | 4 | Раздел 1. Верификация и валидация как часть жизненного цикла программных систем. | 33 | 2 | 1 | 1 | 31 | 25 | 25 | Домашнее задание |
| 2 | 4 | Раздел 2. Формальные методы верификации программного обеспечения. | 16 | 2 | 1 | 1 | 14 | 25 | 25 | Домашнее задание |
| 2 | 4 | Раздел 3. Экспертные методы верификации и валидации в жизненном цикле программных систем. | 27 | 2 | 1 | 1 | 25 | 25 | 25 | Домашнее задание |
| 2 | 4 | Раздел 4. Современные и перспективные подходы к автоматизации процессов верификации программных систем. | 32 | 2 | 1 | 1 | 30 | 25 | 25 | Домашнее задание |
| Всего за 4 семестр | | | 108 | 8 | 4 | 4 | 100 | 100 | 100 | |
| Всего по дисциплине | | | 108 | 8 | 4 | 4 | 100 | 100 | 100 | |

Критерии оценивания

ПСК-2.2

- Вопросы открытого типа:*
- № 1 Валидация программной системы это?
- № 2 Что такое артефакт в программном продукте?
- № 3 Какие инструменты используются для валидации параллельных программ?
- № 4 Какие преимущества предоставляет использование статического анализа для верификации программ?
- № 5 Дополните предложение:
- _____ - это возможность контроля и управления системой для поддержания стабильности и производительности.
- № 6 Дополните предложение:
- Основное преимущество автоматизации верификации заключается в _____ затрат времени на проверку качества ПО?
- № 7 Какие методы применяются при автоматизации процессов верификации в программной инженерии?
- № 8 Что используется для создания математических моделей программ при формальной верификации?
- № 9 Что представляют собой формальные методы в верификации ПО
- № 10 Каков основной результат использования формальных методов верификации?
- Вопросы закрытого типа:*
- № 1 Что из перечисленного описывает процесс валидации?
- Проверка соответствия спецификациям
- Поиск и исправление ошибок
- Оценка производительности программы
- Подтверждение соответствия требованиям пользователя
- № 2 Что такое нагрузочное тестирование?
- Тестирование системы на устойчивость к непредвиденным ситуациям
- Тестирование системы на корректную работу с большими объемами данных
- Тестирование всей системы в целом, как правило, через ее пользовательский интерфейс
- № 3 Что определяют варианты использования?
- Как функции, так и требования
- Только функции системы
- Только требования к системе
- № 4 Установите соответствие определений:
1. Устойчивость к отказам
2. Согласованность данных
3. Эффективность передачи данных
- А- Способность системы продолжать работу при отказе отдельных компонентов.
- Б- Уровень эффективности передачи информации между узлами системы
- В- Гарантия того, что данные системы остаются консистентными и корректными при изменениях.
- № 5 Установите последовательность по порядку повышения производительности ПО в

| | |
|------|--|
| | параллельных системах: |
| | А) Распределение нагрузки |
| | Б) Анализ узких мест |
| | В) Оптимизация кода |
| | Г) Оптимизация ввода-вывода |
| | Д) Кэширование данных |
| № 6 | Е) Использование параллельных алгоритмов Установите соответствие определений: |
| | 1. Отзывчивость |
| | 2. Балансировка нагрузки |
| | 3. Безопасность |
| | А- Равномерное распределение нагрузки между узлами системы для оптимального использования ресурсов. |
| | Б- Защита от несанкционированного доступа, обеспечение конфиденциальности данных. |
| № 7 | В- Скорость реакции системы на запросы пользователей или изменения в окружении Что включает в себя организационная экспертиза в рамках специализированных экспертных методов? |
| | Анализ пользовательского интерфейса |
| | Оценка организационной структуры |
| | Аудит безопасности |
| № 8 | Эвристическая оценка Что представляет собой эвристическая оценка пользовательского интерфейса? |
| | Статистический анализ интерфейса |
| | Экспертное обсуждение интерфейса |
| | Анализ функциональности системы |
| № 9 | Проведение юзабилити-тестирования Какой метод используется для оценки защищенности программных систем на предмет уязвимостей? |
| | Аудит безопасности |
| | Оценка производительности |
| | Исследование рынка |
| № 10 | Анализ пользовательского интерфейса Что включает в себя аудит защищенности программных систем? |
| | Оценка организационной структуры |
| | Анализ пользовательского интерфейса |
| | Проверка системы на уязвимости |

Вопросы открытого типа:

- № 1 Дополните предложение:
- _____ - это деятельность, направленная на выяснение их правильности или, наоборот, ошибочности
- № 2 Что такое модель жизненного цикла?
- № 3 Чем характеризуется устойчивость программного обеспечения?
- № 4 Дополните предложение:
- _____ программной системы - это структура и организация компонентов программы, которая определяет ее функциональность и взаимодействие между ними.
- № 5 Какие аспекты включает в себя процесс валидации параллельных программных систем?
- № 6 Что означает понятие "тестирование программного обеспечения"?
- № 7 Какие методы используются для верификации параллельных программ?
- № 8 Какие факторы влияют на сложность валидации распределенных систем?
- № 9 Дополните предложение:
- Автоматизация процессов верификации упрощает _____ дефектов.
- № 10 Дополните предложение:
- Современные подходы к автоматизации верификации включают в себя _____ анализ.
- Вопросы закрытого типа:*
- № 1 Где осуществляется верификация программной системы?
- На стадии проектирования
- На стадии тестирования
- На стадии разработки
- На стадии сопровождения
- № 2 Какова основная цель валидации программной системы?
- Устранение ошибок кода
- Проверка соответствия требованиям заказчика
- Улучшение производительности системы
- Оценка качества кодирования
- № 3 Что включает в себя процесс верификации?
- Подтверждение соответствия требованиям
- Оценку производительности системы
- Поиск и исправление дефектов
- Работу с пользовательскими интерфейсами
- № 4 Какой из процессов оценивает, правильно ли созданная система решает задачи своего предназначения?
- Верификация
- Валидация
- Тестирование
- Разработка

| | |
|------|--|
| № 5 | <p>Что такое управление версиями:</p> <p>Автоматизированный процесс трансформации исходных текстов ПО в пакет исполняемых модулей</p> <p>Хранение нескольких версий одного и того же документа</p> <p>Ручной процесс трансформации исходных текстов ПО в пакет исполняемых модулей</p> |
| № 6 | <p>При выполнении какого вида тестирования система тестируется на устойчивость к непредвиденным ситуациям?</p> <p>При выполнении нагрузочного тестирования</p> <p>При выполнении интеграционного тестирования</p> <p>При выполнении стрессового тестирования</p> |
| № 7 | <p>При использовании какого метода тестирования код программы доступен тестирующим?</p> <p>При использовании любого метода тестирования</p> <p>При использовании метода белого ящика</p> |
| № 8 | <p>При использовании метода черного ящика</p> <p>Установите соответствие определений:</p> <ol style="list-style-type: none"> 1. Надежность 2. Масштабируемость 3. Производительность <p>А- Эффективность выполнения задач системой в различных условиях нагрузки.</p> <p>Б- Гарантия корректной работы системы в условиях отказов и ошибок.</p> <p>В- Способность системы эффективно обрабатывать увеличение объема данных или пользовательской нагрузки.</p> |
| № 9 | <p>Установите последовательность этапов оценки качества ПО в распределенных системах:</p> <p>А) Анализ характеристик</p> <p>Б) Определение требований</p> <p>В) Мониторинг</p> <p>Г) Тестирование</p> |
| № 10 | <p>Д) Оптимизация</p> <p>Установите последовательность этапов обеспечения безопасности в распределенных системах:</p> <p>А) Идентификация уязвимостей</p> <p>Б) Улучшение защиты по мере необходимости</p> <p>В) Реализация защитных мер</p> <p>Г) Мониторинг и аудит</p> <p>Д) Разработка политики безопасности</p> |