

УТВЕРЖДАЮ
Декан факультета

_____ Матвеев П.В.

« ____ » _____ 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: 2025

Программу составил:

Кафедра О7 Информационные системы и программная инженерия
Бухарин Владимир Владимирович, д.т.н., 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:

ОПК-4 — Способен применять на практике новые научные принципы и методы исследований

ПК-2.2 — Способен обеспечить управление архитектурой интегрированного программного обеспечения и единой информационной среды

Формированию компетенций служит достижение следующих результатов образования:

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;

навыки:

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

ПК-2.2

знания:

основ иерархии информационных систем в части уровней "данные - информация - знания";

умения:

выбирать и применять технологию искусственного интеллекта для решения или поддержки решения поставленной задачи;

навыки:

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: **РАЗРАБОТКА И ОФОРМЛЕНИЕ ТЕХНИЧЕСКОЙ ДОКУМЕНТАЦИИ НА ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ**.

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

Предварительные компетенции, сформированные у обучающегося до начала изучения дисциплины:

- ОПК-3 — Способен анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями

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		ОПК-4	ПК-2.2
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.	32	2	1	1	30	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.	33	3	1	2	30	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.	22	2	1	1	20	25	25
2	4	Раздел 4. Статические и динамические методы верификации программных систем. 4.1. Статический анализ артефактов жизненного цикла. Проверка выполнения правил корректности построения артефактов. 4.2. Поиск типичных ошибок и дефектов в артефактах жизненного цикла статическими методами. 4.3. Классификация методов динамической верификации. 4.4. Реальная и имитационная динамическая верификация. 4.5. Характеристики качества, оцениваемые динамическими методами. 4.6. Верификационный мониторинг программных систем. Профилирование как частный случай мониторинга. 4.7. Тестирование как динамический метод верификации программных систем. Связь тестирования с валидацией и отладкой программных систем.	21	1	1	0	20	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.	Описание процессов в исчислении взаимодействующих систем. Спецификация протокола как системы взаимодействующих агентов	0.5
4		Вычисление бисимуляции и определение вида эквивалентности программных агентов	0.5
5		Верификация поведения программных агентов формальными методами (контрольная работа №1)	1
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
Всего за 4 семестр			4

3.3. Самостоятельная работа студента (СРС)

№	Номер и наименование раздела	Содержание учебного задания	Объем,
---	------------------------------	-----------------------------	--------

п/п	дисциплины		часов
1	Раздел 1. Верификация и валидация как часть жизненного цикла программных систем.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	15
2		Подготовка к практическим занятиям	15
3	Раздел 2. Формальные методы верификации программного обеспечения.	Подготовка к контрольной работе	7
4		Подготовка к практическим занятиям	7
5		Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	7
6		Выполнение домашнего задания	9
7	Раздел 3. Экспертные методы верификации и валидации в жизненном цикле программных систем.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	10
8		Подготовка к практическим занятиям	10
9	Раздел 4. Статические и динамические методы верификации программных систем.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	10
10		Подготовка к практическим занятиям	10
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			Контр.Р.	ДР					ДЗ	ДР	Контр.Р., ДЗ, зач.

Условные обозначения:

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

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

- диагностическая работа;
- контрольная 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, 62 экз.
2. В. Н. Каминский. . Веб-программирование. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014, 94 экз.
3. И. С. Петухов. . Разработка программного обеспечения. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007, 66 экз.
4. И. С. Петухов. . Разработка программного обеспечения. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007, эл. рес.
5. Н. Н. Смирнова. . Верификация и тестирование программных систем. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014, 20 экз.
6. Н. Н. Смирнова. . Верификация и тестирование программных систем. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014, эл. рес.
7. Н. Н. Смирнова, С. Д. Тарасов. . Основы построения компиляторов. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007, эл. рес.
8. С. В. Сеницын, Н. Ю. Налютин. . Верификация программного обеспечения. М.: Интернет-Ун-т Информ. Технол., 2008, 5 экз.
9. 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:

1. Прикладная информатика.

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

1. <https://e.lanbook.com/> — ЭБС Лань;
2. <https://ibooks.ru/> — ЭБС Айбукс.ру - это большой выбор актуальной литературы для вашей библиотеки в электронном виде;
3. http://library.voenmeh.ru/jirbis2/index.php?option=com_irbis&view=irbis&Itemid=474 — Фундаментальная библиотека БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова;
4. <https://urait.ru/> — Главная – Образовательная платформа Юрайт. Для вузов и ссузов.;
5. <http://www.emis.de/ELibM.html> - Electronic Library of Mathematics..

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:

не требуется.

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. Проектор;
2. Аудитория с числом посадочных мест не меньше количества обучающихся.

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 Естественнотехнический БГТУ "ВОЕНМЕХ" им. Д.Ф. Устинова кафедрой О7 Информационные системы и программная инженерия.

Дисциплина нацелена на формирование *компетенций*:

ОПК-4 Способен применять на практике новые научные принципы и методы исследований;

ПК-2.2 Способен обеспечить управление архитектурой интегрированного программного обеспечения и единой информационной среды.

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;
- контрольная 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 дидактических единиц по рекомендуемой литературе	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) С. В. Синицын, Н. Ю. Налютин. . Верификация программного обеспечения: М.: Интернет-Ун-т Информ. Технол., 2008 (1.1-1.3, 1.6-1.8)	15
Подготовка к практическим занятиям	С. М. Старолетов. . Основы тестирования и верификации программного обеспечения: Санкт-Петербург: Лань, 2020 (1.1)	15
Итого по разделу 1		30
Раздел 2. Формальные методы верификации программного обеспечения.		
Подготовка к контрольной работе	И. С. Петухов. . Разработка программного обеспечения: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007 (1.2)	7
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)	7
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	С. М. Старолетов. . Основы тестирования и верификации программного обеспечения: Санкт-Петербург: Лань, 2020 (2.1) С. В. Синицын, Н. Ю. Налютин. . Верификация программного обеспечения: М.: Интернет-Ун-т Информ. Технол., 2008 (1.1-1.3, 1.6-1.8)	7
Выполнение домашнего задания		9
Итого по разделу 2		30
Раздел 3. Экспертные методы верификации и валидации в жизненном цикле программных систем.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	. Управление программными проектами: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2013 (Теоретические сведения к лабораторным работам №1 и №3)	10
Подготовка к практическим занятиям	С. В. Синицын, Н. Ю. Налютин. . Верификация программного обеспечения: М.: Интернет-Ун-т Информ. Технол., 2008 (1.5, 1.8-1.10, 6.1, 6.2, 9, 10) И. С. Петухов. . Разработка программного обеспечения: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007 (1.1) Н. Н. Смирнова. . Верификация и тестирование	10

	программных систем: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1.1)	
Итого по разделу 3		20
Раздел 4. Статические и динамические методы верификации программных систем.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	. Управление программными проектами: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2013 (Теоретические сведения к лабораторным работам №2-4) Н. Н. Смирнова, С. Д. Тарасов. . Основы построения компиляторов: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007 (2-4) В. Н. Каминский. . Веб-программирование: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (Раздел «Регулярные выражения»)	10
Подготовка к практическим занятиям	С. М. Старолетов. . Основы тестирования и верификации программного обеспечения: Санкт-Петербург: Лань, 2020 (1.3, 1.4)	10
Итого по разделу 4		20

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

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

- диагностическая работа
- домашнее задание;
- контрольная работа;
- 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.

Контрольная работа

Контрольная работа №1 включает в себя четыре задания: два теоретических вопроса на знание понятийного аппарата формальных методов верификации и две задачи – одна на описание агентов в исчислении взаимодействующих систем и одна на вычисление бисимуляции и определение вида эквивалентности агентов. Для получения оценки «удовлетворительно» необходимо полное и правильное решение хотя бы одной задачи или полный развернутый ответ на хотя бы один теоретический вопрос. Более высокая оценка формируется с учетом результатов выполнения остальных заданий.

Контрольная работа №2 включает в себя четыре задания: три теоретических вопроса на знание понятийного аппарата статических, динамических и синтетических методов верификации программных систем и одну задачу на выбор метода или группы верификации исходя из конкретной задачи верификации. Для получения оценки «удовлетворительно» необходимо полное и обоснованное решение задачи или полный развернутый ответ на хотя бы один теоретический вопрос. Более высокая оценка формируется с учетом результатов выполнения остальных заданий.

Вопросы к контрольным работам и примеры ситуативных задач приведены в УМК дисциплины.

Зачет

Итоговый тест считается успешно пройденным при выполнении с рейтингом не менее 70% (получении не менее 70 баллов из 100). Тест состоит из 10 вопросов, на выполнение теста отводится 15 минут.

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		ОПК-4	ПК-2.2	
2	4	Раздел 1. Верификация и валидация как часть жизненного цикла программных систем.	32	2	1	1	30	25	25	Домашнее задание, Контрольная работа
2	4	Раздел 2. Формальные методы верификации программного обеспечения.	33	3	1	2	30	25	25	Домашнее задание, Контрольная работа
2	4	Раздел 3. Экспертные методы верификации и валидации в жизненном цикле программных систем.	22	2	1	1	20	25	25	Контрольная работа
2	4	Раздел 4. Статические и динамические методы верификации программных систем.	21	1	1	0	20	25	25	Контрольная работа
Всего за 4 семестр			108	8	4	4	100	100	100	
Всего по дисциплине			108	8	4	4	100	100	100	

**Оценочные материалы по дисциплине ТЕОРИЯ ВЕРИФИКАЦИИ И ВАЛИДАЦИИ
ПАРАЛЛЕЛЬНЫХ И РАСПРЕДЕЛЕННЫХ ПРОГРАММНЫХ СИСТЕМ**

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

№ 1 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Какие классы языков спецификации задач существуют?

Запишите номера выбранных ответов без пробелов и точек и обоснование выбора

1. Языки описания требований
2. Языки описания задач
3. Языки функциональных спецификаций
4. Языки функциональной реализации

№ 2 Прочитайте текст и установите соответствие

Установите соответствие распределенных систем из левого столбца по принципу векторных параллельных процессоров и симметричных из правого столбца.

К каждой позиции в левом столбце, подберите позицию из правого столбца.

- | | | |
|----|------------------------|------------------------|
| 1. | Соединенной памятью | А. Мультипроцессоры |
| 2. | Разделяемой памятью | Б. Мультикомпьютеры |
| 3. | Объединенной памятью | В. Компьютеры |
| 4. | Распределенной памятью | Г. Таких не существует |
| | | Д. Майнфреймы |

№ 3 Прочитайте текст и установите соответствие

Для классов архитектур параллельных компьютеров в левом столбце, подберите их обозначение из правого столбца.

К каждой позиции в левом столбце, подберите позицию из правого столбца.

- | | | |
|----|---|--|
| 1. | один поток команд и один поток данных | А. SIMD (Single Instruction stream/Multiple Data stream) |
| 2. | один поток команд и множество потоков данных | Б. MISD (Multiple Instruction stream/Single Data stream) |
| 3. | множество потоков команд и один поток данных | В. SISD (Single Instruction stream/Single Data stream) |
| 4. | множество потоков команд и множество потоков данных | Г. DDS (Direct Digital Synthesis) |
| | | Д. MIMD (Multiple Instruction stream/Multiple Data stream) |

№ 4 Прочитайте текст и установите последовательность

Выберете правильную последовательность основных стадий жизненного цикла разработки программного обеспечения

Запишите соответствующую последовательность цифр слева направо без пробелов и точек.

1. Тестирование продукта
2. Разработка продукта
3. Проектирование архитектуры продукта
4. Развертывание и обслуживание
- 5 Планирование и анализ требований
6. Определенные требования

№ 5 Прочитайте текст и запишите развернутый обоснованный ответ

Одним из статистических методов верификации программных систем являются статические анализаторы программ. Охарактеризуйте статические анализаторы программ?

№ 6 Прочитайте текст и запишите развернутый обоснованный ответ

Основная цель параллельных вычислений является уменьшение времени решения задачи и одним из подходов – увеличивать мощность процессорных устройств. Какие есть ограничения при таком подходе?

№ 7 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

В каком виде при верификации записываются требования из каталога с использованием математического аппарата?

Запишите номера выбранных ответов без пробелов и точек и обоснование выбора

1. Формальной спецификацией
2. Концептуальной моделью
3. Формальной моделью
4. Частной спецификацией

№ 8 Прочитайте текст и установите последовательность

Выберете правильную последовательность спиральной модели этапов жизненного цикла разработки программного обеспечения

Запишите соответствующую последовательность цифр слева направо без пробелов и точек.

1. Планирование
2. Разработка и верификация
3. Определение целей
4. Оценка альтернатив

№ 9 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Определите какие функции не используются в тестовых сценариях.

1. Все системные функции, доступные через меню
2. Комбинации функций, доступные через меню
3. Функции с неправильным вводом данных
4. Функции с правильным выводом данных

№ 10 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Какой принцип не используется при разработки программного обеспечения методом "чистая комната"?

1. Формальная спецификация
2. Пошаговое тестирование системы
3. Статистическое тестирование системы
4. Статическая верификация

№ 11 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Что проверяется при структурного тестирования ветвей?

1. Все ветви программы выполняется хотя бы один раз
2. Каждая зависимая ветвь программы выполняется хотя бы один раз
3. Каждая независимая ветвь программы выполняется хотя бы один раз
4. Каждая независимая ветвь программы выполняется хотя бы несколько раз

№ 12 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Укажите три основных типа поддержки программного обеспечения?

1. Корректирующее обслуживание
2. Корпоративное обслуживание
3. Адаптивное обслуживание
4. Развивающая поддержка

ПК-2.2 - Способен обеспечить управление архитектурой интегрированного программного обеспечения и единой информационной среды

№ 1 Прочитайте текст и запишите развернутый обоснованный ответ

Важнейшими качествами функциональной спецификаций являются однозначность и понятность. Исходя из указанных качеств, функциональных спецификаций, какими свойствами должны обладать языки функциональных спецификаций?

№ 2 Прочитайте текст и запишите развернутый обоснованный ответ

Планирование испытаний необходимо при процессе разработки и тестирования. Дайте краткое описание данного процесса?

№ 3 Прочитайте текст и установите соответствие

Для каждого языка функциональных спецификаций в левом столбце, подберите из правого столбца средств описания предметной области.

К каждой позиции в левом столбце, подберите позицию из правого столбца.

1. Табличные языки

2. Языки равенств

Средство
описания
экспертных
систем
операционной

- А. семантикой,
получаемое
введем
определенных
правил
подстановки
- Б. Записи пред- и
постусловий,
преобразователей

3.	Логические языки	В. Исходный машинный или промежуточный код
4.		Г. Двухместная функция или структуры данных

№ 4 Прочитайте текст и установите соответствие

Для каждого названия этапа статического анализа в левом столбце, подберите его содержание из правого столбца.

К каждой позиции в левом столбце, подберите позицию из правого столбца.

1.	Анализ потока управления	А. проверяется использование переменных в программе
2.	Анализ использования данных	Б. проверяется согласованность различных частей программы, правильность объявления процедур и их использования
3.	Анализ интерфейса	В. идентифицируются и выделяются циклы, их точки входа и выхода, а также неиспользуемый код
4.	Анализ потоков данных	Г. определяются все ветви программы и выделяются операторы, исполняемые в каждой ветви
5.		Д. определяются зависимости между исходными (входными) и результирующими (выходными) переменными

№ 5 Прочитайте текст и установите последовательность

Расположите в правильной последовательности фазы классической водопадной модели

Запишите соответствующую последовательность цифр слева направо без пробелов и точек.

1. Системное тестирование
2. Разработка продукта
3. Анализ проекта

4. Поддержка

5. Проектирование продукта

6. Анализ требований

№ 6 Прочитайте текст и установите последовательность

Расположите в правильной последовательности основные этапы верификации

Запишите соответствующую последовательность цифр слева направо без пробелов и точек.

1.Формализация требований

2.Анализ результатов

3.Анализ документации

4. Извлечение тестов

№ 7 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Что достигается при валидации программных систем?

Запишите номер выбранного ответа без точки и обоснование выбора

1. проверка правильности программного кода

2. проверка соответствия требований к программной системе

3. установление наличия функций в программе

4. проверка скорости выполнения программы

№ 8 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

При каком методе тестирования требуется достаточно большое время для начала сборки модулей?

Запишите номер выбранного ответа без точки и обоснование выбора

1. Восходящий метод

2. Нисходящий метод

3. Метод большого скачка

4. Метод сэндвича

№ 9 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Какая запись соответствует предусловию заданной функции в спецификации программы при аксиоматическом подходе верификации?

1. $pre-f(d)$

2. $post-f(d,r)$

3. $f(t1,...,tk)$

№ 10 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Укажите три основных подхода к представлению семантики утверждений?

Запишите номера выбранных ответов без пробелов и точек и обоснование выбора

1. операционный

2. денотационный

3. алгоритмический

4. аксиоматический

№ 11 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Укажите что включает в себя внешняя спецификация?

Запишите номера выбранных ответов без пробелов и точек и обоснование выбора

1. Функциональную спецификацию

2. Функциональную ведомость исполнения

3. Множество правил взаимодействия

4. Аксиоматические теории проблемной области

№ 12 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Какие методы тестирования используются при сборки объектно-ориентированных систем?

Запишите номера выбранных ответов без пробелов и точек и обоснование выбора

1. Тестирование потоков

2. Тестирование структуры

3. Тестирование взаимодействий между объектами

4. Тестирование памяти