
УТВЕРЖДЕНА

Решением Ученого совета
АНО ВО «Центральный университет»
«25» мая 2026 г.
Протокол №2

**Рабочая программа дисциплины (модуля)
«Инструменты формальной верификации»**

Направление подготовки: 02.03.01 Математика и компьютерные науки

Направленность (профиль) подготовки: Прикладной искусственный интеллект

Квалификация (степень) выпускника: бакалавр

Форма обучения: очная

Срок освоения программы: 4 года

Год набора: 2026

**Москва
2026**

Содержание

1. Краткая характеристика дисциплины (модуля)	3
2. Перечень планируемых результатов обучения	5
3. Тематический план	7
4. Содержание дисциплины (модуля)	8
5. Учебно-методическое обеспечение	9
6. Материально-техническое обеспечение	9
7. Методические и оценочные материалы	11

1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «Инструменты формальной верификации» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – бакалавриат по направлению 02.03.01 «Математика и компьютерные науки», профиль «Прикладной искусственный интеллект», утвержденный приказом Министерства науки и высшего образования Российской Федерации № 807 от 23.08.2017 года.

Изучение дисциплины (модуль) «Инструменты формальной верификации» позволяет освоить навыки работы инженером-верификатором, разработчиком статических анализаторов, консультантом по безопасности, исследователем в области формальных методов.

Место дисциплины (модуля) в структуре образовательной программы

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки бакалавриата по направлению 02.03.01 «Математика и компьютерные науки», профиль «Прикладной искусственный интеллект» и входит в вариативную часть Блока 1, формируемую участниками образовательных отношений, как дисциплина по выбору.

Дисциплина (модуль) доступна для изучения на 3 или 4 курсе в 6, 7, 8 семестрах на выбор после успешного освоения дисциплин (модулей): «Математический анализ. Часть 1», «Математический анализ. Часть 2» и «Линейная алгебра и геометрия. Часть 1», «Линейная алгебра и геометрия. Часть 2» или «Математический анализ. Продвинутый уровень. Часть 1», «Математический анализ. Продвинутый уровень. Часть 2» и «Линейная алгебра и геометрия. Продвинутый уровень. Часть 1», «Линейная алгебра и геометрия. Продвинутый уровень. Часть 2».

Цель изучения дисциплины (модуля): сформировать у студентов теоретические знания и практические навыки применения формальных методов для доказательства корректности программных и аппаратных систем и интеграции этих процессов в современный цикл разработки.

Задачи изучения дисциплины (модуля):

- освоить основы формализма – модели (Kripke-структуры, переходные системы), логики спецификации (LTL, CTL, μ -калькул, Hoare);
- изучить основные методы верификации – модель-чеккинг, автоматные доказательства, абстрактная интерпретация и др;
- овладеть современными инструментами (SPIN, NuSMV, PRISM, UPPAAL, CBMC, KLEE, Z3, Coq, Isabelle и т.п.);
- научиться интерпретировать результаты – анализ контр-примеров, локализация причин ошибок, оценка покрываемости.

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

- основные классы инструментов формальной верификации: SMT-решатели, model checkers, proof assistants, static analyzers и средства символьного исполнения;
- основные виды проверяемых свойств: предусловия, постусловия, инварианты, assertions, safety- и liveness-свойства;
- принципы работы SAT/SMT-решателей и их применение к проверке логических ограничений;
- базовые идеи model checking, анализа достижимости и проверки темпоральных свойств;
- назначение proof assistants и отличие механизированного доказательства от автоматического поиска контрпримеров;

уметь:

- формулировать проверяемые свойства для небольших программ, алгоритмов и моделей систем;
- использовать SMT-решатель для проверки выполнимости ограничений и поиска контрпримеров;
- выполнять базовое символьное исполнение программы и строить path conditions;
- строить простые конечные модели систем и проверять их инварианты с помощью model checking;
- читать и писать простые формальные доказательства или контракты в выбранном инструменте верификации;

владеть:

- навыками выбора подходящего инструмента формальной верификации под конкретную задачу;
- приёмами записи спецификаций, assertions, инвариантов и условий корректности;
- базовыми навыками работы с SMT/model checking/proof assistant инструментами;
- навыками анализа контрпримеров и исправления спецификаций или программ на их основе;
- навыками оценки ограничений инструментов верификации: неполноты, масштабируемости, ложных срабатываний и сложности формализации.

2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

Код и содержание компетенции	Индикатор компетенции и перечень планируемых результатов обучения по дисциплине (модулю)
УК-1. Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1. Знает методы поиска и анализа информации в области разработки, основные принципы критической оценки источников информации и их релевантности.
	УК-1.2. Умеет критически оценивать источники информации и синтезировать данные из различных источников для решения задач, применять системный подход к анализу и решению комплексных проблем.
	УК-1.3. Имеет практический опыт работы с современными инструментами и технологиями для обработки информации, формулировании и структурировании задач на основе полученной информации.
УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1. Знает действующие правовые нормы, регулирующие деятельность в области решения задач, основные методы и подходы к определению круга задач.
	УК-2.2. Умеет определять круг задач в рамках поставленной цели, выбирать оптимальные способы решения задач, учитывая имеющиеся ресурсы и ограничения.
	УК-2.3. Имеет практический опыт применения знаний о правовых нормах и ресурсах в реальных ситуациях, разработки и реализации решений в соответствии с установленными ограничениями.
ОПК-1. Способен консультировать и использовать фундаментальные знания в области математического анализа, комплексного и функционального анализа алгебры, аналитической геометрии, дифференциальной геометрии и топологии, дифференциальных уравнений, дискретной математики и математической логики, теории вероятностей, математической статистики и случайных процессов, численных методов, теоретической механики в профессиональной деятельности	ОПК-1.1. Знает основные понятия, методы и результаты математического анализа, комплексного и функционального анализа, алгебры, аналитической и дифференциальной геометрии, топологии, дифференциальных уравнений, дискретной математики и математической логики, теории вероятностей, математической статистики и случайных процессов, численных методов и теоретической механики.
	ОПК-1.2. Умеет применять фундаментальные математические знания и методы для анализа и решения профессиональных задач.
	ОПК-1.3. Имеет практический опыт выбора, разработки и реализации математических моделей и методов при решении задач профессиональной деятельности.
ОПК-6. Способен разрабатывать алгоритмы и компьютерные программы, пригодные для	ОПК-6.1. Знает принципы алгоритмизации, основные структуры данных, парадигмы программирования, методы разработки, тестирования и отладки компьютерных программ.
	ОПК-6.2. Умеет разрабатывать, документировать, тестировать и

практического применения	отлаживать алгоритмы и компьютерные программы, пригодные для практического применения, с использованием современных языков и сред разработки.
	ОПК-6.3. Имеет практический опыт разработки и сопровождения компьютерных программ и программных комплексов для решения прикладных задач.

3. Тематический план

№п/ п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы				ТКУ (текущий контроль успеваемости)
		Очная форма				
		Аудиторная работа		Контроль	Самостоятельная работа	
		Лекции	Практические занятия			
1	Введение в инструменты верификации	4	4		30	Домашнее задание Аудиторная работа
2	SMT-решатели и логические ограничения	6	6		30	Домашнее задание Аудиторная работа
3	Проверка моделей	10	10	2	30	Домашнее задание Контрольная работа Аудиторная работа
4	Proof assistants и механизированные доказательства	10	12	2	30	Домашнее задание Контрольная работа Аудиторная работа
	Зачет с оценкой			4		Коллоквиум
Итого:		30	32	8	120	
Объем дисциплины (модуля) (в ак. ч.)		190				
Объем дисциплины (модуля) (в зач. ед.)		5				

4. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	Введение в инструменты верификации	Ошибки в программных системах. Ограничения тестирования. Формальная верификация как способ проверки свойств для всех возможных входов или состояний. Спецификация программы. Различие между свойствами функции, алгоритма и системы состояний
2	SMT-решатели и логические ограничения	Задача SAT. Выполнимость булевых формул. SMT как расширение SAT Основы работы с Z3. Переменные, ограничения, модели. Поиск значений, удовлетворяющих системе ограничений. SMT для проверки программных свойств Идея symbolic execution. Символьные входы. Path conditions. Ветвления и ограничения путей. Генерация тестов и поиск ошибок. Статический анализ программ. Over-approximation и under-approximation. Абстрактная интерпретация на концептуальном уровне. Анализ интервалов, возможных значений и недостижимого кода. Контрольная работа 1
3	Проверка моделей	Проверка моделей. Состояния, переходы, начальные состояния, плохие состояния. Проверка достижимости. Контрпример как путь в системе состояний. Интуиция LTL/CTL. Свойства вида «никогда не происходит ошибка», «событие когда-нибудь произойдет», «после запроса будет ответ». TLA+ или аналогичный инструмент для спецификации систем. Переменные состояния, действия, инварианты, next-state relation. Проверка моделей на малых конфигурациях. Коллоквиум
4	Proof assistants и механизированные доказательства	Интерактивные системы доказательств: Lean, Coq, Isabelle. Доказательства как проверяемые объекты. Различие между автоматическим решателем и proof assistant. Термы, типы, утверждения, тактики. Простые доказательства равенств, импликаций, конъюнкций и свойств функций. Роль библиотек. Формальная спецификация функций над списками, деревьями, натуральными числами. Индукция как основной метод доказательства свойств рекурсивных программ. Контрактное программирование и специализированные инструменты. Контрольная работа 2

5. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый студент в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Студентам обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

Основная литература:

1. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем : учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2025. — 432 с. — (Высшее образование). — ISBN 978-5-534-07604-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/561885>

2. Трубочкина, Н. К. Основы технологии производства и машинное обучение : учебник для вузов / Н. К. Трубочкина. — Москва : Издательство Юрайт, 2026. — 383 с. — (Высшее образование). — ISBN 978-5-534-22010-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/600548>

Дополнительная литература:

1. Набатова, Д. С. Математические и инструментальные методы поддержки принятия решений : учебник и практикум для вузов / Д. С. Набатова. — Москва : Издательство Юрайт, 2026. — 292 с. — (Высшее образование). — ISBN 978-5-534-02699-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/583202>

6. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

- столами и стульями;
- компьютерной техникой;
- механическими калькуляторами;
- специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и

обеспечением доступа в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека elibrary.ru библиотека	https://elibrary.ru/defaultx.asp
2.	База данных для IT-специалистов	https://habr.com
3.	База данных ScienceDirect	https://www.sciencedirect.com
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
5.	Федеральный портал «Российское образование»	https://www.edu.ru/
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
7.	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
8.	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модуле), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
Операционные системы:		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное
Браузеры:		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
Офисные приложения:		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
Программное обеспечение для планирования и учета времени:		
Toggle app	зарубежное	свободно распространяемое
Системы управления проектами:		
Microsoft Imagine (Project)	зарубежное	лицензионное
Системы управления базами данных:		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
Системы резервного копирования (backup):		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
Справочно-правовые системы:		
КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
Средства антивирусной защиты:		

Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
Среды разработки:		
Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое
CopelliaSim	зарубежное	свободно распространяемое
Google Colaboratory	зарубежное	свободно распространяемое
Пакеты программных средств и библиотек:		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
Системы управления библиографической информацией:		
Zotero	зарубежное	свободно распространяемое
Сервисы и службы:		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое

7. Методические и оценочные материалы

Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Инструменты формальной верификации» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекция, семинары, домашние задания, аудиторная работа, контрольные работы, коллоквиум, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

Лекция — систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удастся, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

Семинар — это форма учебной деятельности, проводимая в учебном заведении под руководством преподавателя, где студенты активно участвуют в обсуждениях, практических заданиях и других формах взаимодействия.

Для успешной подготовки к семинару рекомендуется заранее ознакомиться с темой занятия и основными материалами, чтобы иметь возможность активно участвовать в обсуждении. Также полезно подготовить вопросы и идеи для обсуждения, что поможет глубже понять материал и продемонстрировать заинтересованность.

Аудиторная работа — активная работа студента на семинаре или лекции, его ответы на вопросы преподавателя и участие в дискуссии.

Для успешного участия в аудиторной работе студентам рекомендуется заранее ознакомиться с темой обсуждения, прочитать необходимые материалы и подготовить вопросы. Важно активно слушать и вовлекаться в дискуссию, высказывая свои мнения, аргументируя их. При ответах на вопросы преподавателя стоит быть уверенным, четким и

логичным, опираясь на изученный материал. Также полезно поддерживать диалог с однокурсниками, чтобы обогатить обсуждение и расширить свои знания.

Домашнее задание – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал. Использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

Контрольная работа – письменная работа с набором задач, которые нужно решить за ограниченное время.

Цель контрольной работы - получить специальные знания по одной или нескольким темам дисциплины (модуля) и продемонстрировать навыки их практического применения.

Коллоквиум – устные ответы на вопросы, список которых известен студенту заранее.

В процессе подготовки к коллоквиуму необходимо проанализировать учебные материалы, ознакомившись с лекциями, учебниками и дополнительными источниками, акцентируя внимание на ключевых темах. Рекомендуется создать структурированные конспекты, выделяя основные идеи, термины и формулы.

Самостоятельная работа – работа студентов, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

Система оценивания результатов обучения по дисциплине (модулю)

Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «Инструменты формальной верификации»

Оценивание уровня учебных достижений, обучающихся по дисциплине (модулю), осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

Промежуточная аттестация по дисциплине (модулю) осуществляется в форме **зачета с оценкой**, при этом проводится оценка компетенций, сформированных по дисциплине.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом, если иное не указано в силлабусе дисциплины:

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
10	Отлично	Зачтено	Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину. Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать,
9	Отлично	Зачтено	
8	Отлично	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
			конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами.
7	Хорошо	Зачтено	Студент обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
6	Хорошо	Зачтено	
5	Удовлетворительно	Зачтено	Студент обладает базовыми знаниями по дисциплине (модулю),
4	Удовлетворительно	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
			но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
3	Не сдан	Не зачтено	Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	Не зачтено	
1	Не сдан	Не зачтено	

Дисциплина (модуль) «Инструменты формальной верификации» оценивается следующим образом:

Активность	Вес	Количество	Описание
Домашние задания	25%	10	Набор задач по темам недели
Аудиторная работа	15%	15	Активная работа студента на занятии
Контрольные работы	35%	2	Письменная работа с набором задач, которые нужно решить за ограниченное время
Зачет с оценкой	20%	1	Коллоквиум - устные ответы на вопросы, список которых известен студенту заранее

Формула расчёта итоговой оценки по дисциплине (модулю) «Инструменты формальной верификации»: $\langle 0,25 \times \text{среднее за домашние задания} + 0,15 \times \text{среднее за аудиторную работу} + 0,35 \times \text{среднее за контрольные работы} + 0,2 \times \text{зачет с оценкой} \rangle$.

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные вопросы для подготовки к занятию

1. Что такое **Kripke-модель** ? Какой набор элементов её определяет?
2. Чем отличаются **LTL** и **CTL** по способу описания времени? Приведите по одному оператору-примеру.
3. Что такое **state-space explosion** ? Какие три техники обычно применяются для её уменьшения?
4. Дайте определение **SMT-solver** в одном-двух предложениях.
5. Перечислите три самых популярных инструмента **model-checking** и укажите, для каких типов моделей каждый из них оптимален.
6. Что такое **инвариант** в стиле Хоара? Как он проверяется статическим анализом?
7. Как формулируется свойство **жизнеспособности** (liveness) в CTL?

8. Что представляет собой **bounded model checking** (BMC) и в чём его основное отличие от обычного model checking?
9. Какие типы **контр-примеров** выдаёт инструмент **SPIN** ?
10. Что такое **abstract interpretation** ? Какой элемент решётки обычно выбирают для анализа целочисленных переменных?

Примерные домашние задания

1. **Модель-чеккинг в NuSMV**
 - Сформулируйте модель простого протокола «клиент-сервер» (запрос → ответ → готовность).
 - Запишите два свойства:
 1. *Безопасность* : «не может быть одновременно два клиента в состоянии «ожидание ответа».
 2. *Жизнеспособность* : «каждый запрос в конечном итоге получает ответ».
 - Запустите проверку, представьте файл модели `.smv` , файл свойств `.spec` и скриншоты/логи результатов. В случае обнаружения нарушения объясните причину.
2. **SMT-решение в Z3 (Python API)**
 - Сформулируйте задачу: найдите целые x, y, z такие, чтобы $x + 2*y - z = 7$ и $x*y > 10$.
 - Напишите скрипт, который либо возвращает модель, либо доказывает отсутствие решения.
 - Комментируйте полученный результат (что он говорит о совместимости ограничений).
3. **Формальная верификация функции сортировки**
 - Реализуйте в C функцию `int insertion_sort(int *a, int n)`.
 - Добавьте в код **assert**-выражения, проверяющие отсутствие выхода за границы массива и сохранение перестановки элементов.
 - Примените **CBMC** с параметрами `--bounds-check --unwind 20`.
 - Составьте короткий отчёт (не более 1 стр.) о найденных ошибках (если они есть) и о том, какие дополнительные развертки требуются для полной проверки.
4. **Интерактивное доказательство в Coq**
 - Доказать лемму о сложении натуральных чисел: `forall n m, n + S m = S (n + m)`.
 - Использовать минимум два тактика (например, `induction, simpl`).
 - Предоставить файл `plus_succ.v` и скриншот успешного выполнения `coqc`.

Примерные задания для контрольной работы

Часть А – теоретические вопросы (каждому 3 балла)

1. Дайте определение **Kripke-модели** (одно предложение).
2. Назовите два основных типа свойств, проверяемых в model checking, и укажите пример формулы для каждого в LTL.
3. Что такое **bounded model checking** ? Перечислите два ограничения, которые он налагает на проверяемую модель.
4. Объясните, как **SMT-solver** сочетает несколько теорий (пример: линейная арифметика + булева логика).
5. Перечислите три техники борьбы с «state-space explosion» и укажите, к какой группе (символические/абстрактные/другие) относится каждая из них.

Часть В – задачи с небольшим кодом (каждой 7 баллов)

1. **Model checking в SPIN**
 - Напишите модель простого «producer-consumer» (одно-producer, один-consumer) в PROMELA (не более 80 строк).
 - Запишите свойство «никогда не происходит падения потребителя, когда очередь пуста» в виде LTL-выражения.

— С помощью SPIN проверьте свойство; если нарушено, укажите контр-пример (путь).

2. SMT-задача в Z3

— Сформулируйте в Z3 задачу: найти целые **a, b, c** такие, чтобы $a^2 + b^2 = c^2$, $a + b + c = 30$.

— Приведите короткий скрипт (≤ 15 строк) и результат его выполнения.

3. Формальная проверка функции в C

— Дана функция `int pow2(int x) { return x * x; }`. Добавьте **assert**-проверки, гарантирующие отсутствие переполнения 32-разрядного знакового целого.

— Примените **SBMC** и укажите, найдено ли переполнение. Если да – приведите конкретные значения **x**.

Часть С – открытый вопрос (10 баллов)

Опишите процесс интеграции формального верификационного шага в **CI/CD-pipeline** на примере проекта, использующего GitHub Actions. Укажите, какие артефакты (модели, спецификации, отчёты) должны храниться, как обрабатывать неуспешные проверки и какие метрики (время проверки, покрытие свойств) можно мониторить.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция	Индикатор
1.	Дайте определение формальной верификации в отличие от тестирования.	Формальная верификация доказывает корректность для всех возможных входов математически, тестирование проверяет только выборочные сценарии	ОПК-1	ОПК-1.1
2.	Назовите три основных вида спецификаций свойств системы.	Свойства безопасности (ничего плохого не случится), свойства живости (что-то хорошее случится), инварианты	ОПК-1	ОПК-1.1
3.	Объясните разницу между спецификацией функции и спецификацией системы состояний.	Спецификация функции описывает вход-выход, спецификация системы описывает поведение во времени и переходы между состояниями	УК-1	УК-1.2
4.	В чем основное различие между задачами SAT и SMT?	SAT решает выполнимость булевых формул, SMT решает выполнимость формул с теориями (арифметика, массивы, битовые векторы)	ОПК-1	ОПК-1.1
5.	Опишите роль SMT-решателя Z3 в проверке программных свойств.	Z3 проверяет выполнимость логических ограничений, полученных из кода, и находит контрпримеры или доказывает их отсутствие	ОПК-6	ОПК-6.2

6.	Что такое символьное выполнение (symbolic execution)?	Метод анализа, где программа выполняется с символьными переменными вместо конкретных значений, собирая ограничения на путях	ОПК-6	ОПК-6.1
7.	Объясните понятие path condition в символьном выполнении.	Логическое условие, которое должно быть истинным для достижения определенной точки программы при заданных символьных входах	ОПК-1	ОПК-1.2
8.	Дайте определение абстрактной интерпретации в статическом анализе.	Метод приближенного вычисления семантики программы через абстрактные домены (интервалы, знаки) для гарантии сходимости	ОПК-1	ОПК-1.1
9.	Опишите структуру модели Крипке для model checking.	Кортеж из множества состояний, отношений переходов, множества атомарных пропозиций и функции маркировки состояний	ОПК-1	ОПК-1.1
10.	Запишите формулу LTL для свойства никогда не происходит ошибка.	$G \neg \text{error}$ (Глобально не ошибка) или $\neg \text{error}$ в нотации темпоральной логики	ОПК-6	ОПК-6.2
11.	Запишите формулу CTL для свойства существует путь где eventualmente P.	$EF P$ (Exists Finally P) в стандартной нотации CTL	ОПК-1	ОПК-1.2
12.	Что такое контрпример в контексте проверки моделей?	Последовательность состояний (трасса), демонстрирующая нарушение специфицированного свойства системы	УК-1	УК-1.2
13.	Назовите инструмент для спецификации распределенных систем на основе темпоральной логики действий.	TLA+ (Temporal Logic of Actions) или язык спецификаций PlusCal	ОПК-6	ОПК-6.2
14.	В чем различие между автоматическим решателем и интерактивным proof assistant?	Решатель работает автономно и выдает да/нет, ассистент требует руководства пользователя и строит проверяемое доказательство	УК-2	УК-2.2
15.	Назовите две популярные системы интерактивных доказательств.	Coq, Lean, Isabelle, Agda (любые две из перечисленных)	ОПК-1	ОПК-1.1
16.	Какую роль играют тактики в proof assistants?	Тактики — это команды, которые преобразуют текущую цель	ОПК-6	ОПК-6.1

		доказательства в подцели, автоматизируя шаги вывода		
17.	Почему индукция является основным методом доказательства в proof assistants?	Индукция позволяет доказывать свойства для бесконечных множеств (натуральные числа, списки) через базу и шаг рекурсии	ОПК-1	ОПК-1.2
18.	Объясните концепцию типы как утверждения в соответствии Карри-Ховарда.	Тип программы соответствует логическому утверждению, а терм (код) программы соответствует доказательству этого утверждения	ОПК-1	ОПК-1.1
19.	Как формальная верификация интегрируется в CI/CD пайплайны?	Автоматический запуск статических анализаторов, SMT- проверок или генерации тестов при каждом коммите кода	УК-2	УК-2.2
20.	Назовите основное ограничение применения формальных методов в индустрии.	Высокая трудоемкость, необходимость экспертизы и сложность масштабирования на большие legacy системы	УК-1	УК-1.3