Федеральное государственное бюджетное образовательное
учреждение высшего образования
Московский государственный университет имени М.В. Ломоносова
Механико-математический факультет

УТВЕРЖДАЮ
декан механико-
математического факультета
_____/А.И.
Шафаревич /
«_14_» _октября__2021г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Наименование дисциплины:

Математические методы верификации схем и программ

Уровень высшего образования:
магистратура

Направление подготовки / специальность:

02.04.01 "Математика и компьютерные науки" (3++)

Направленность (профиль)/специализация ОПОП:

Интеллектуальные системы. Теория и приложения

Форма обучения:
очная

Рабочая программа рассмотрена и одобрена

на заседании Ученого совета механико- математического факультета

(протокол № 7 от 14 октября 2021 года)

Москва 2021

Рабочая программа дисциплины (модуля) разработана в соответствии с Федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) для реализуемых основных профессиональных образовательных программ высшего образования по направлению подготовки 02.04.01 "Математика и компьютерные науки" утвержденного Приказом Министерства образования и науки РФ от 10 января 2018 г. N 13.

1. Место дисциплины (модуля) в структуре ОПОП ВО:
дисциплина относится к блоку профессиональной подготовки вариативной части ОПОП ВО.

2. Входные требования для освоения дисциплины (модуля), предварительные условия (если есть):
Для того чтобы изучение дисциплины было возможно, обучающийся должен обладать следующими компетенциями:
Знать: основные понятия, концепции, результаты и методы дискретной математики, программирования и математической логики.
Уметь: решать стандартные задачи дискретной математики и математической логики.

3. Результаты обучения по дисциплине (модулю), соотнесенные с требуемыми компетенциями выпускников.

Планируемые результаты обучения по дисциплине (модулю)		
Содержание и код компетенции.	**Индикатор (показатель) достижения компетенции**	**Планируемые результаты обучения по дисциплине, сопряженные с индикаторами достижения компетенций**
ПК-4. Способен адаптировать и применять методы и алгоритмы машинного обучения для решения прикладных задач в различных предметных областях	ПК-4.1. Ставит задачи по адаптации или совершенствованию методов и алгоритмов для решения комплекса задач предметной области	ПК-4.1. З-1. Знает классы методов и алгоритмов машинного обучения ПК-4.1. У-1. Умеет ставить задачи и адаптировать методы и алгоритмы машинного обучения

4. Объем дисциплины (модуля) составляет 3 з.е., в том числе 48 академических часов, отведенных на контактную работу обучающихся с преподавателем, 60 академических часов на самостоятельную работу обучающихся.

5. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий:
5.1. Структура дисциплины (модуля) по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий (в строгом соответствии с учебным планом)

Наименование разделов и тем дисциплины (модуля), **Форма промежуточной аттестации по дисциплине (модулю)**	**Номинальные трудозатраты обучающегося**			**Всего академических часов**	**Форма текущего контроля успеваемости* (наименование)**
	Контактная работа (работа во взаимодействии с преподавателем) Виды контактной работы, академические часы		**Самостоятельная работа обучающегося, академические часы**		
	Занятия лекционного типа	Занятия семинарского типа			
Тема 1 Задача верификации информационных систем и общие подходы к ее решению	6	3	10	28	опрос
Тема 2 Табличные, символьные и теоретико-автоматные методы верификации моделей программ	12	6	24	52	опрос
Тема 3 Методы верификации информационных систем реального времени	8	4	16	34	опрос
Тема 4 Метод повышения эффективности алгоритмов верификации	6	3	10		
Другие виды самостоятельной работы (отсутствуют)	—	—			—
Промежуточная аттестация (экзамен)					
Итого	*32*	*16*	*60*	108	—

5.2. Содержание разделов (тем) дисциплины

№ п/п	Наименование разделов (тем) дисциплины	Содержание разделов (тем) дисциплин	
1.	Тема 1. Задача верификации информационных систем и общие подходы к ее решению	Задача верификации аппаратуры и программного обеспечения. Зачем нужна формальная верификация программ. Основные подходы к задаче формальной верификации. Принципы верификации моделей программ. Исторические сведения. Достижения методов формальной верификации программ. Алгоритмические и комбинаторные трудности применения метода верификации моделей программ. Общие принципы дедуктивной верификации программ. Операционная семантика императивных программ. Формальная постановка задачи верификации программ. Логика Хоара: правила вывода и свойства. Автоматизация проверки правильности программ. Моделирование схем. Системы переходов - модели Крипке. Представление систем переходов формулами логики предикатов первого порядка. Синхронные и асинхронные схемы. Степень детализации представления. Трансляция описаний программ и схем в модели Крипке	24
2.	Тема 2. Табличные, символьные и теоретико-автоматные методы верификации моделей программ	Темпоральная логика деревьев вычислений CTL. Синтаксис и семантика CTL. Примеры спецификаций моделей в терминах формул CTL. Темпоральная логика линейного времени PLTL. Синтаксис и семантика PLTL. Свойства живости и безопасности. Ограничения справедливости. Задача верификации моделей (model-checking). Табличный алгоритм верификации моделей для CTL. Обоснование корректности и сложности табличного алгоритма верификации моделей. Проблема “комбинаторного взрыва”. Символьные средства описания моделей. Двоичные разрешающие диаграммы (BDD). Алгоритм редукции BDD к каноническому виду (OBDD). Выполнение операций над OBDD: унарные и бинарные булевы операции, квантификация, проверка выполнимости, подсчет числа единиц. Общие представления о сложности в классе OBDD Представления неподвижной точки в CTL. Алгоритм символьной верификации моделей в CTL. Табличная верификация моделей для PLTL. Обобщенные автоматы Бюхи, трансляция формул LTL в автоматы. Сведение задачи проверки выполнимости формул PLTL к проблеме пустоты для автоматов Бюхи. Алгоритм двойного поиска в глубину с возвратом (DDFS) для проверки пустоты автомата Бюхи.	

3.	Тема 3. Методы верификации информационных систем реального времени.	Временные автоматы как формальные модели распределенных систем реального времени. Вычисления временных автоматов. Примеры использования временных автоматов для моделирования встроенных систем. Зеноновские вычисления. Синтаксис и семантика Timed CTL. Примеры формальных спецификаций поведения встроенных систем при помощи TCTL Задача верификации моделей программ реального времени. Отношение эквивалентности часов и регионы. Регионные системы переходов. Оценка числа регионов. Сведение задачи верификации временных автоматов относительно TCTL к задаче верификации моделей Крипке относительно CTL
4.	Тема 4. Метод повышения эффективности алгоритмов верификации.	Отношения бисимуляционной эквивалентности (бисимуляции) и симуляционного квазипорядка (симуляции) на моделях Крипке. Равновыполнимость темпоральных формул на бисимуляционно эквивалентных моделях Крипке. Вычисление классов бисимуляционной эквивалентности на конечных моделях Крипке. Упрощение моделей Крипке при помощи отношений симуляции и и бисимуляции. Редукция моделей Крипке по конусу влияния. Абстракции данных при построении моделей Крипке. Верификация моделей программ для вычислений ограниченной длины (bounded model checking, BMC). Сведение задачи BMC к задаче проверки выполнимости булевых формул (SAT). Применение автоматических средств решения задачи SAT для решения задачи BMC

6. Фонд оценочных средств (ФОС, оценочные и методические материалы) для оценивания результатов обучения по дисциплине (модулю).

6.1. Типовые контрольные задания или иные материалы для проведения текущего контроля успеваемости, критерии и шкалы оценивания (в отсутствие утвержденных соответствующих локальных нормативных актов на факультете)

Примеры заданий для практических занятий

1. Доказательство корректности императивных программ с помощью логики Хоара. Методы построения инвариантов программ.
Типовая задача. Записать в виде предусловия и постусловия требование корректности программы, записанное на естественном языке
а). программа записывает в переменную prod произведение значений x и y
б). программа записывает в переменные quo, rem частное и остаток от деления положительного значения x на положительное значение y
в). программа меняет местами значения переменных x, y
г). программа записывает в переменную N наибольший общий делитель значений x, y
д). программа записывает в переменную m максимальный элемент непустого массива s[0 : n−1]
е). программа разворачивает непустой массив s[0 : n−1] задом наперёд
2. Устройство и возможности практического применения пакеты построения и преобразования ROBDD CUDD
Типовая задача. Построить ROBDD для заданного порядка переменных, реализующую ту же функцию, что и заданная формула.
3. Устройство программно-инструментального средства верификации моделей программ SMV. Язык описания моделей и задания спецификаций в системе SMV. Примеры применения системы SMV на практике. Верификации простых моделей с использованием системы SMV: описание моделей, формальное задание спецификаций, проверка выполнимости спецификаций.
4. Язык описания систем взаимодействующих процессов Promela. Примеры описаний распределенных систем. Примеры применения с программно-инструментального средства верификации моделей программ SPIN на практике. Верификации простых моделей с использованием системы SPIN: описание моделей, формальное задание спецификаций, проверка выполнимости спецификаций.
5. Язык описания сетей конечных временных автоматов в программно-инструментальном средстве верификации моделей программ UPPAAL. Примеры описаний сетей конечных временных автоматов. Язык запросов системы UPPAAL. Верификация простых сетей временных автоматов при помощи средства верификации UPPAAL.

6.2. Типовые контрольные задания или иные материалы для проведения промежуточной аттестации по дисциплине, критерии и шкалы оценивания

Программа экзамена:

1. Основные методы верификации аппаратуры и программного обеспечения – тестирование, имитационное моделирование, дедуктивный анализ, верификация моделей.
2. Преимущества метода верификации моделей. Алгоритмические и комбинаторные трудности применения метода верификации моделей.
3. Общие принципы дедуктивной верификации программ.

4. Операционная семантика императивных программ.
5. Логика Хоара: правила вывода и свойства. Формальная постановка задачи верификации программ.
6. Автоматизация проверки правильности программ.
7. Моделирование схем. Системы переходов (модели Крипке).
8. Представление систем переходов формулами логики предикатов первого порядка.
9. Синхронные схемы. Моделирование электронных схем.
10. Асинхронные схемы. Моделирование параллельных программ.
11. Временные автоматы. Моделирование информационных систем реального времени.
12. Формальные языки спецификации моделей. Построение модели автомата (протокола, управляющего алгоритма) на языках описания моделей программ (SMV, Promela, сети временных автоматов).
13. Двоичные разрешающие диаграммы (BDD). Алгоритм редукции BDD к каноническому виду (ROBDD).
14. Выполнение операций над ROBDD: унарные и бинарные Булевы операции, операция ITE (мультиплексорная функция от трех переменных), квантификация, проверка выполнимости, подсчет числа единиц.
15. Эффективная машинная реализация ROBDD на основе хэш-таблиц. Общие представления о сложности в классе ROBDD (зависимость сложности от порядка переменных, сложность умножения целых чисел).
16. Реализация алгоритмов работы с ROBDD на примере одного из распространенных пакетов (CUDD, ABCD, и др.).
17. Конъюнктивные нормальные формы (CNF). Задачи выполнимости КНФ. Сведение задачи выполнимости булевой формулы (или схемы) к задаче выполнимости КНФ.
18. Алгоритм DPLL. Эвристические методы повышения производительности на примере существующего SAT-солвера (Chaff, BerkMin, MiniSat, etc.) Схемные SAT-солверы (решение задачи выполнимости схемы без сведения к КНФ).
19. Темпоральная логика деревьев вычислений CTL. Синтаксис и семантика CTL. Примеры спецификаций моделей в терминах формул CTL.
20. Темпоральная логика линейного времени PLTL. Синтаксис и семантика PLTL.
21. Свойства живости и безопасности.
22. Ограничения справедливости.
23. Задача верификации моделей (model-checking).
24. Табличный алгоритм верификации моделей для CTL.
25. Обоснование корректности и сложности табличного алгоритма верификации моделей.
26. Проблема “комбинаторного взрыва”.
27. Представления неподвижной точки.
28. Алгоритм символьной верификации моделей для CTL.
29. Особенности реализации алгоритма: учет ограничений справедливости, расщепленные отношения переходов, рекомбинация произведений.
30. Табличная верификация моделей для PLTL.
31. Обобщенные автоматы Бюхи, трансляция формул LTL в автоматы.
32. Сведение задачи проверки выполнимости формул PLTL к проблеме пустоты для автоматов Бюхи.
33. Алгоритм двойного поиска в глубину с возвратом (DDFS) для проверки пустоты автомата Бюхи.

34. Временные автоматы как формальные модели распределенных систем реального времени. Вычисления временных автоматов.
35. Примеры использования временных автоматов для моделирования встроенных систем. Задача верификации временных автоматов. Зеноновские вычисления.
36. Синтаксис и семантика Timed CTL. Примеры формальных спецификаций поведения встроенных систем при помощи TCTL. Задача верификации моделей программ реального времени.
37. Отношение эквивалентности часов и регионы. Регионные системы переходов. Оценка числа регионов.
38. Сведение задачи верификации временных автоматов относительно TCTL к задаче верификации моделей Крипке относительно CTL.
39. Отношения бисимуляционной эквивалентности (бисимуляции) и симуляционного квазипорядка (симуляции) на моделях Крипке.
40. Равновыполнимость темпоральных формул на бисимуляционно эквивалентных моделях Крипке.
41. Вычисление классов бисимуляционной эквивалентности на конечных моделях Крипке.
42. Упрощение моделей Крипке при помощи отношений симуляции и и бисимуляции.
43. Редукция моделей Крипке по конусу влияния.
44. Абстракции данных при построении моделей Крипке
45. Верификация моделей программ для вычислений ограниченной длины (bounded model checking, BMC).
46. Сведение задачи BMC к задаче проверки выполнимости булевых формул (SAT). Применение автоматических средств решения задачи SAT для решения задачи BMC.

РЕЗУЛЬТАТ ОБУЧЕНИЯ	**КРИТЕРИИ и ПОКАЗАТЕЛИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ из соответствующих карт компетенций**					**ОЦЕНОЧНЫЕ СРЕДСТВА**
	1	2	3	4	5	
	Неудовлетворительно	**Неудовлетворительно**	**Удовлетворительно**	**Хорошо**	**Отлично**	
З1 Знать: методы построения формальных моделей программ и описаний информационных систем, выразительные возможности темпоральных логик,	Отсутствие знаний	Фрагментарные представления о методах построения формальных моделей программ и описаниях информационных систем, выразительных возможностях темпоральных логик, используемых в качестве языков спецификации	В целом сформированные, но неполные знания о методах построения формальных моделей программ и описаниях информационных систем, выразительных возможностях темпоральных логик, используемых в качестве языков	Сформированные, но содержащие отдельные пробелы знания о методах построения формальных моделей программ и описаниях информационных систем, выразительных возможностях темпоральных логик,	Сформированные систематические знания о методах построения формальных моделей программ и описаниях информационных систем, выразительных возможностях темпоральных логик,	индивидуальное собеседование

используемых в качестве языков спецификации распределенных программ и описаний информационных систем, алгоритмы верификации формальных моделей распределенных программ и описаний информационных систем		распределенных программ и описаний информационных систем, алгоритмов верификации формальных моделей распределенных программ и описаний информационных систем	спецификации распределенных программ и описаний информационных систем, алгоритмов верификации формальных моделей распределенных программ и описаний информационных систем	используемых в качестве языков спецификации распределенных программ и описаний информационных систем, алгоритмов верификации формальных моделей распределенных программ и описаний информационных систем	используемых в качестве языков спецификации распределенных программ и описаний информационных систем, алгоритмов верификации формальных моделей распределенных программ и описаний информационных систем	
У1 Уметь правильно записывать темпоральные спецификации распределенных программ и описаний информационных систем, использовать методы и алгоритмы верификации формальных моделей программ	Отсутствие умений	Фрагментарные умения правильно записывать темпоральные спецификации распределенных программ и описаний информационных систем, использовать методы и алгоритмы верификации формальных моделей программ	В целом сформированное, но не систематическое умение правильно записывать темпоральные спецификации распределенных программ и описаний информационных систем, использовать методы и алгоритмы верификации формальных моделей программ	Сформированное, но содержащее отдельные пробелы умение правильно записывать темпоральные спецификации распределенных программ и описаний информационных систем, использовать методы и алгоритмы верификации формальных моделей программ	Сформированное систематическое умение правильно записывать темпоральные спецификации распределенных программ и описаний информационных систем, использовать методы и алгоритмы верификации формальных моделей программ	практическое контрольное задание

В1 Владеть навыками использования системы верификации моделей программ SMV, SPIN и UPPAAL	Отсутствие навыков	Фрагментарное владение навыками использования системы верификации моделей программ SMV, SPIN и UPPAAL	В целом сформированное, но не систематическое владение навыками использования системы верификации моделей программ SMV, SPIN и UPPAAL	Сформированное, но содержащее отдельные пробелы владение навыками использования системы верификации моделей программ SMV, SPIN и UPPAAL	Сформированное систематическое владение навыками использования системы верификации моделей программ SMV, SPIN и UPPAAL	практическое контрольное задание

7. Ресурсное обеспечение:
7.1. Перечень основной и дополнительной литературы
Перечень основной и дополнительной учебной литературы:

1) Ю.Г. Карпов. Model checking: верификации параллельных и распределенных программных систем. Изд-во БХВ-Петербург, 2010, 552 с.
2) Christel Baier. Joost-Pieter Katoen. Principles of Model Checking. The MIT Press. Cambridge, Massachusetts. London, England, 2008, 980 pp
3) Э.М. Кларк, О. Грамберг, Д. Пелед. «Верификация моделей программ». Москва, 2002, изд-во МЦНМО, 415 с.
4) M. R. A. Huth, M.D. Ryan. Logic in Computer Science: Modelling and Reasoning about Systems. Cambridge University Press, 2002, 387 p.
5) Kenneth L. McMillan, Interpolation and SAT-Based Model Checking. Proceedings of CAV 2003, p. 1-13.
6) Karl S. Brace, Richard L. Rudell and Randal E. Bryant. Efficient Implementation of a BDD Package. In Proceedings of the 27th ACM/IEEE Design Automation Conference (DAC 1990), pages 40–45. IEEE Computer Society Press, 1990.
7) Daniel Kroening, Ofer Strichman. Decision Procedures. Springer, 2008, 304 p.

7.2. Перечень лицензионного программного обеспечения, в том числе отечественного производства

1. NuSMV: a new symbolic model checker. http://nusmv.fbk.eu/.
2. CUDD: CU Decision Diagram Package. http://vlsi.colorado.edu/~fabio/CUDD/.
3. SPIN: http://spinroot.com/spin/whatispin.html

Материально-техническая база

7.3. Перечень профессиональных баз данных и информационных справочных систем
Нет

7.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. https://mk.cs.msu.ru/

Описание материально-технического обеспечения.

Учебная аудитория, оснащенная мультимедийными средствами демонстрации.
Пакет программ CUDD построения и преобразования ROBDD
Программно-инструментальное средство верификации моделей программ NuSMV
Программно-инструментальное средство верификации моделей программ SPIN
Программно-инструментальное средство верификации моделей программ UPPAAL

8. Соответствие результатов обучения по данному элементу ОПОП результатам освоения ОПОП указано в Общей характеристике ОПОП.

9. Разработчик (разработчики) программы.

д.ф.- м.н., профессор Захаров Владимир Анатольевич (zakh@cs.msu.su)
к.ф.-м.н., доцент Подымов Владислав Васильевич (valdus@yandex.ru)