

**Федеральное государственное автономное образовательное
учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»**

УТВЕРЖДЕНО

**Директор физтех-школы
прикладной математики и
информатики**

А.М. Райгородский

	Рабочая программа дисциплины (модуля)
по дисциплине:	Автоматизация программирования
по направлению:	Прикладная математика и информатика
профиль подготовки:	А1360: Передовые методы искусственного интеллекта Физтех-школа Прикладной Математики и Информатики кафедра алгоритмов и технологий программирования
курс:	4
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 8 (весенний) - Дифференцированный зачет

Аудиторных часов: 30 всего, в том числе:

лекции: 15 час.

семинары: 15 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 15 час.

Всего часов: 45, всего зач. ед.: 1

Программу составил: А.С. Хританков, канд. физ.-мат. наук, доцент

Программа обсуждена на заседании кафедры алгоритмов и технологий программирования 12.02.2024

Аннотация

Дисциплина "Автоматизация программирования" предусматривает изучение студентами современных методов разработки и проектирования программных систем и программно-аппаратных комплексов, основанных на применении комплексных моделей систем, генераторов программ и предметно-ориентированных языков моделирования.

1. Цели и задачи

Цель дисциплины

Дать развернутое представление о проблематике, фундаментальных концепциях, принципиальных возможностях, современном состоянии и тенденциях развития технологий разработки программного обеспечения с использованием моделей.

Задачи дисциплины

Выработать навыки работы с CASE-инструментами разработки программных систем с применением технологий MDE/MDD.

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.1 Способен применять современные вычислительную технику и сервисы сети Интернет в области (сфере) профессиональной деятельности
	ОПК-2.2 Знает и умеет применять численные математические методы и прикладное программное обеспечение для решения научных задач в профессиональной области
	ОПК-2.3 Знает основные требования информационной безопасности
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации
	ПК-2.2 Способен планировать и проводить научные исследования самостоятельно или в качестве члена (руководителя) малого научного коллектива
	ПК-2.3 Способен проводить апробацию результатов научно-исследовательской работы посредством публикации научных статей и участия в конференциях

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны

знать:

- теоретические основы разработки программного обеспечения на основе моделей;
- универсальный язык моделирования UML 2, базовый и основной UML (fUML), язык действий для UML (ALF);
- технологию исполняемого UML (Executable UML), типовые решения и принципы;
- методы преобразования моделей, методы отображения моделей в текст;
- технологию предметно-ориентированной разработки, типовые решения и принципы;
- способы применения технологий MDD в промышленной разработке ПО;
- методы тестирования, основанного на моделях.

уметь:

- ориентироваться в технологиях разработки на основе моделей, определять подходящую технологию для решения прикладных задач;
- разрабатывать предметно-ориентированные языки, трансляторы с них;
- исследовать предметную область и разрабатывать модели предметной области.

владеть:

- инструментами разработки с использованием моделей Rational Rhapsody, Eclipse DSL.

4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа
1	Введение в технологии программной инженерии на основе моделей (MDD).	2	2		2
2	Технологии исполняемого UML.	2	2		2
3	Методы обработки моделей в MDD.	2	2		2
4	Предметно-ориентированные языки (DSL).	2	2		2
5	Применения MDSD.	2	2		2
6	Автоматизация разработки Automation-in-the-Small.	2	2		2
7	Тестирование и верификация программных систем на основе моделей (MBT).	3	3		3
Итого часов		15	15		15
Подготовка к экзамену		0 час.			
Общая трудоёмкость		45 час., 1 зач.ед.			

4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 8 (Весенний)

1. Введение в технологии программной инженерии на основе моделей (MDD).

Моделирование, метамоделирование. Определение MDSD, основные направления и области применения. Поколения инструментов.

2. Технологии исполняемого UML.

Основы моделирования с UML. Foundational UML (fUML), Action Language for fUML (ALF). Технология ExecutableUML, методы проектирования в OOA/RD и ExecutableUML.

3. Методы обработки моделей в MDD.

Генерация, трансляция, интерпретация. Технологии преобразования Model-to-Model и отображения в текст Model-to-Text. Алгоритмические, алгебраические и графовые модели преобразования.

4. Предметно-ориентированные языки (DSL).

Введение в Domain-Driven Design. Понятие о DSL, основные виды и паттерны разработки DSL. Методология DDD по Эвансу, эвристики и паттерны. Повсеместный язык (Ubiquitous language).

5. Применения MDSD.

Фабрики программного обеспечения Software Factories. Концепция Microsoft. Примеры Cargemini и TCS. Разработка программных систем в Motorola. Встраиваемое ПО. Линейки программных продуктов.

6. Автоматизация разработки Automation-in-the-Small.

Методы Automation-in-the-Small. Генерация UI по POJO/POCO. Автоматизация процесса сборки с помощью DSL.

7. Тестирование и верификация программных систем на основе моделей (MBT).

Введение в автоматизированное тестирование. Методы записи-воспроизведения, скрипты тестирования, табличные и DSL методы. Тестирование как сравнение с моделью. Применение и виды моделей. Критерии генерации и качества тестовых наборов.

5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Учебная аудитория, оснащенная компьютером и мультимедийным оборудованием (проектор, звуковая система).

6. Перечень рекомендуемой литературы

Основная литература

Программирование на C++ [Электронный ресурс], Электрон. версия печ. публикации / Н. Дейл, Ч. Уимз, М. Хедингтон. — М., ДМК Пресс, 2007

Введение в программирование, учебное пособие / И. Ю. Баженова, В. А. Сухомлин. — Москва, ИНТУИТ, 2016. — URL: <https://e.lanbook.com/book/100695> (дата обращения: 30.12.2020). -

Полный текст (Режим доступа : из сети МФТИ / Удаленный доступ)

Дополнительная литература

Эрик Эванс. Предметно-ориентированное проектирование (DDD). Структуризация сложных программных систем. – М.: Вильямс. – 2018

7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

Не используются

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)

На лекционных занятиях используются мультимедийные технологии, включая демонстрацию презентаций.

9. Методические указания для обучающихся по освоению дисциплины (модуля)

Студент, изучающий дисциплину, должен с одной стороны, овладеть общим понятийным аппаратом, а с другой стороны, должен научиться применять теоретические знания на практике. Успешное освоение дисциплины требует:

- посещения студентом всех видов аудиторных занятий;
- ведения конспекта в ходе лекционных занятий;
- качественной самостоятельной подготовки к практическим занятиям, активной работы на них;
- активной самостоятельной и аудиторной работы студента;

- своевременной сдачи преподавателю заданий по аудиторным видам работ.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению: Прикладная математика и информатика
профиль подготовки: AI360: Передовые методы искусственного интеллекта
Физтех-школа Прикладной Математики и Информатики
кафедра алгоритмов и технологий программирования
курс: 4
квалификация: бакалавр

Семестр, формы промежуточной аттестации: 8 (весенний) - Дифференцированный зачет

Разработчик: А.С. Хританков, канд. физ.-мат. наук, доцент

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.1 Способен применять современные вычислительную технику и сервисы сети Интернет в области (сфере) профессиональной деятельности
	ОПК-2.2 Знает и умеет применять численные математические методы и прикладное программное обеспечение для решения научных задач в профессиональной области
	ОПК-2.3 Знает основные требования информационной безопасности
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации
	ПК-2.2 Способен планировать и проводить научные исследования самостоятельно или в качестве члена (руководителя) малого научного коллектива
	ПК-2.3 Способен проводить апробацию результатов научно-исследовательской работы посредством публикации научных статей и участия в конференциях

2. Показатели оценивания компетенций

В результате изучения дисциплины «Автоматизация программирования» обучающийся должен:

знать:

- теоретические основы разработки программного обеспечения на основе моделей;
- универсальный язык моделирования UML 2, базовый и основной UML (fUML), язык действий для UML (ALF);
- технологию исполняемого UML (Executable UML), типовые решения и принципы;
- методы преобразования моделей, методы отображения моделей в текст;
- технологию предметно-ориентированной разработки, типовые решения и принципы;
- способы применения технологий MDD в промышленной разработке ПО;
- методы тестирования, основанного на моделях.

уметь:

- ориентироваться в технологиях разработки на основе моделей, определять подходящую технологию для решения прикладных задач;
- разрабатывать предметно-ориентированные языки, трансляторы с них;
- исследовать предметную область и разрабатывать модели предметной области.

владеть:

- инструментами разработки с использованием моделей Rational Rhapsody, Eclipse DSL.

3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

1. Моделирование на унифицированном языке моделирования.
2. Технологии автоматизированной разработки ПО – исполняемый UML.
3. Разработка предметно-ориентированного языка и транслятора.
4. Примеры реализации MDSD в индустрии. Пример Motorola.
5. Понятие о фабриках программного обеспечения. Линейка программных продуктов. Схема фабрики.
6. Основные проблемы и критические инновации в фабриках программного обеспечения.
7. Определение тестирования. Виды тестирования. Основные методы: ручной, с записью тестов, разработка скриптов, с применением DSL.
8. Тестирование на основе моделей (MBT). Определение. Метод. Виды моделей.
9. Критерии качества тестового набора.

4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

1. Определение MDSD. Расскажите об основных препятствиях в использовании MDSD. В какой части цикла инноваций для разработки приложений Gartner по вашему мнению находится MDSD?
2. Уровни моделирования в модели MOF от OMG вы знаете? Приведите пример.
3. Поколения инструментов MDSD вы знаете, опишите их, в чем различия? Приведите примеры.
4. Сравните схемы состояний UML 2 и конечные автоматы в ExecutableUML.
5. Расскажите о структуре технологии ExecutableUML, основные элементы моделей, технология генерации приложений.
6. Стандартизация технологии исполняемого UML: fUML и Alf. Текущее состояние и реализация.
7. Подходы в трансформации моделей. Программный, шаблонный (text-to-text), трансляция, преобразования на графах.
8. Методы преобразования графов (Graphrewriting). Алгебраический SPO и DPO. Понятие о TripleGraphGrammars.
9. Определение и примеры DSL. Процесс разработки. Мета модель ECore.
10. Технология разработки DSL. Генераторы и трансляция моделей. Паттерны. Инструменты.
11. Разработка DSL. Абстрактный и конкретный синтаксис. Пример на xText.
12. Определение DDD и его место в процессе разработки. Повсеместный язык (ubiquitous language). Архитектурный каркас.
13. Основные элементы и паттерны DDD.

Критерии оценивания

Оценка "Отлично" (10) - полностью и вовремя решены все задачи без ошибок. Продемонстрирован грамотный подход к решению задач, реализованы оптимальные алгоритмы, код оформлен в едином удобочитаемом стиле.

Оценка "Отлично" (9) - полностью и вовремя решены все задачи без ошибок. Продемонстрирован грамотный подход к решению задач, реализованы оптимальные алгоритмы.

Оценка "Отлично" (8) - полностью и вовремя решены все задачи без ошибок. Продемонстрирован грамотный подход к решению задач.

Оценка "Хорошо" (7) - полностью решены все задачи. Допущены несущественные ошибки.

Оценка "Хорошо" (6) - полностью решено большинство задач. В некоторых задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.

Оценка "Хорошо" (5) - полностью решено две трети задач. В некоторых задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.

Оценка "Удовлетворительно" (4) - полностью решено более половины задач. В остальных задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.

Оценка "Удовлетворительно" (3) - полностью решено более половины задач.

Оценка "Неудовлетворительно" (2) - решено менее половины задач.

Оценка "Неудовлетворительно" (1) - не решено ни одной задачи.

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Дифференцированный зачет может проводиться по итогам текущей успеваемости и сдачи заданий и других видов работ, предусмотренных программой дисциплины и (или) путем организации специального опроса, проводимого в устной и (или) письменной форме.

При проведении устного дифференцированного зачета обучающемуся предоставляется 30 минут на подготовку. Опрос обучающегося не должен превышать одного астрономического часа.

Во время проведения дифференцированного зачета обучающиеся могут пользоваться программой дисциплины, а также справочной литературой, конспектами лекций или другими материалами.