

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Нижегородский государственный университет
им. Н.И. Лобачевского»**

Институт информационных технологий, математики и механики

УТВЕРЖДЕНО
решением Ученого совета ННГУ
протокол № 10 от 02.12.2024 г.

Рабочая программа дисциплины

Компиляторы

Уровень высшего образования
Бакалавриат

Направление подготовки / специальность
02.03.02 - Фундаментальная информатика и информационные технологии

Направленность образовательной программы
Сопряженная разработка программного и аппаратного обеспечения

Форма обучения
очная

г. Нижний Новгород

2025 год начала подготовки

1. Место дисциплины в структуре ОПОП

Дисциплина Б1.В.ДВ.05.01 Компиляторы относится к части, формируемой участниками образовательных отношений образовательной программы.

2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства	
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	Для текущего контроля успеваемости	Для промежуточной аттестации
ПК-3: Способен создавать и исследовать новые математические модели в естественных науках, промышленности и бизнесе, с учетом возможностей современных информационных технологий и программирования и компьютерной техники	<p>ПК-3.1: Знает методы анализа и исследования математических моделей в области фундаментальной информатики и информационных технологий</p> <p>ПК-3.2: Умеет определять ключевые свойства и ограничения системы</p>	<p>ПК-3.1: Знать: методы концептуального, функционального и логического проектирования, основные этапы разработки компилятора</p> <p>ПК-3.2: Владеть: навыками разработки проекта системы, алгоритмами работы при разработке компилятора</p>	Тест	Зачёт: Контрольные вопросы Задачи

3. Структура и содержание дисциплины

3.1 Трудоемкость дисциплины

	очная
Общая трудоемкость, з.е.	3
Часов по учебному плану	108
в том числе	
аудиторные занятия (контактная работа):	
- занятия лекционного типа	32
- занятия семинарского типа (практические занятия / лабораторные работы)	0
- КСР	1
самостоятельная работа	75
Промежуточная аттестация	0 Зачёт

3.2. Содержание дисциплины

(структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий)

Наименование разделов и тем дисциплины	Всего (часы)	в том числе			
		Контактная работа (работа во взаимодействии с преподавателем), часы из них			Самостоятельная работа обучающегося, часы
		Занятия лекционного типа	Занятия семинарского типа (практические занятия/лабораторные работы), часы	Всего	
0 Ф 0	0 Ф 0	0 Ф 0	0 Ф 0	0 Ф 0	
Введение в тензорные компиляторы	8	2		2	6
Математические основы тензорных вычислений	12	4		4	8
Промежуточное представление (IR) в тензорных компиляторах	13	4		4	9
Оптимизация тензорных вычислений	16	6		6	10
Распределение вычислений	16	4		4	12
Генерация кода для целевых устройств	14	4		4	10
Системы автоматической дифференцировки и их связь с компиляторами	14	4		4	10
Перспективы развития тензорных компиляторов	14	4		4	10
Аттестация	0				
КСР	1			1	
Итого	108	32	0	33	75

Содержание разделов и тем дисциплины

1. Введение в тензорные компиляторы (2 ч)

- Понятие тензорного компилятора и его роль в машинном обучении.
- Отличие от классических компиляторов.
- Современные фреймворки: TensorFlow XLA, PyTorch Glow, TVM.

2. Математические основы тензорных вычислений (3 ч)

- Определение тензоров, операции над ними.
- Графы вычислений и их представление.
- Абстракции для работы с многомерными массивами.

3. Промежуточное представление (IR) в тензорных компиляторах (4 ч)

- Формы IR: высокоуровневое, среднее, низкоуровневое представления.
- MLIR как пример многоуровневого IR.
- Преобразования графов и оптимизация представления.

4. Оптимизация тензорных вычислений (5 ч)

- Упрощение выражений, слияние операций, перестановка измерений.

- Автоматическое распараллеливание и векторизация.
- Оптимизация под целевые устройства: GPU, TPU, FPGA.

5. Распределение вычислений (4 ч)

- Разделение вычислений между устройствами.
- Оптимизация коммуникаций между узлами.
- Поддержка гетерогенных архитектур.

6. Генерация кода для целевых устройств (4 ч)

- Выбор инструкций под конкретную архитектуру.
- Учет особенностей памяти и регистров.
- JIT-компиляция и её роль в тензорных вычислениях.

7. Системы автоматической дифференцировки и их связь с компиляторами (3 ч)

- Обратное распространение ошибок и его представление в компиляторах.
- Автоматическая дифференцировка в IR.
- Интеграция с фреймворками глубокого обучения.

8. Перспективы развития тензорных компиляторов (3 ч)

- Компиляция под специализированные ускорители.
- Объединение различных DSL в единую систему.
- Будущее MLIR и других открытых проектов.

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа обучающихся включает в себя подготовку к контрольным вопросам и заданиям для текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведенным в п. 5.

1. Abadi M., Barham P., Chen J. et al. "TensorFlow: A System for Large-Scale Machine Learning" – USENIX Association, 2016.
2. Roesch J., Lyubomirsky S., Weber M. et al. "A Relay Compiler for the TVM Stack" – arXiv preprint, 2018.

5. Фонд оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю)

5.1 Типовые задания, необходимые для оценки результатов обучения при проведении текущего контроля успеваемости с указанием критериев их оценивания:

5.1.1 Типовые задания (оценочное средство - Тест) для оценки сформированности компетенции ПК-3:

1. Какой из следующих терминов относится к компиляции тензорных вычислений?
 - a) Semantic analysis
 - +b) Tensor compilation

- c) Lexical analysis
- d) Code optimization

2. Какой из следующих элементов представляет собой формальное описание вычислений?

- a) AST
- b) CFG
- +c) MLIR
- d) DFA

3. Какой из следующих подходов используется для упрощения тензорных выражений?

- a) Dead code elimination
- +b) Expression simplification
- c) Register allocation
- d) Parsing

4. Какой из следующих процессов позволяет распределять вычисления между несколькими устройствами?

- a) Instruction selection
- +b) Device scheduling
- c) Optimization
- d) Parsing

Критерии оценивания (оценочное средство - Тест)

Оценка	Критерии оценивания
зачтено	не менее 80% правильных ответов в тесте
не зачтено	менее 80% правильных ответов в тесте

5.2. Описание шкал оценивания результатов обучения по дисциплине при промежуточной аттестации

Шкала оценивания сформированности компетенций

Уровень сформированности компетенций (индикатора достижения компетенций)	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно
	не зачтено			зачтено			
<u>Знания</u>	Отсутствие знаний теоретического материала. Невозможность оценить полноту знаний вследствие отказа обучающегося от	Уровень знаний ниже минимальных требований. Имели место грубые ошибки	Минимально допустимый уровень знаний. Допущено много негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько несуществе	Уровень знаний в объеме, соответствующем программе подготовки. Ошибок нет.	Уровень знаний в объеме, превышающем программу подготовки.

	ответа			ошибок	нных ошибок		
<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки	Продемонстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме	Продемонстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания в полном объеме, но некоторые с недочетами	Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания в полном объеме, но некоторые с недочетами	Продемонстрированы все основные умения. Решены все основные задачи с отдельными и несущественными недочетами, выполнены все задания в полном объеме	Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме без недочетов
<u>Навыки</u>	Отсутствие базовых навыков. Невозможность оценить наличие навыков вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки	Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами	Продемонстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продемонстрированы базовые навыки при решении стандартных задач без ошибок и недочетов	Продемонстрированы навыки при решении нестандартных задач без ошибок и недочетов	Продемонстрирован творческий подход к решению нестандартных задач

Шкала оценивания при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно», продемонстрированы знания, умения, владения по соответствующим компетенциям на уровне выше предусмотренного программой
	отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично».
	очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо»
	хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо».
	удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно».
	плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

5.3 Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения на промежуточной аттестации с указанием критериев их оценивания:

5.3.1 Типовые задания (оценочное средство - Контрольные вопросы) для оценки сформированности компетенции ПК-3

1. Что такое тензорный компилятор? Чем он отличается от традиционного?
2. Какие основные этапы проходят данные при компиляции тензорного кода?
3. Что такое промежуточное представление (IR)?
4. Какие уровни IR существуют в тензорных компиляторах?
5. Что такое MLIR и как он используется в тензорной компиляции?
6. Какие виды оптимизаций применяются в тензорных вычислениях?
7. Что такое векторизация и почему она важна?
8. Как осуществляется распределение вычислений между различными устройствами?
9. Как работает JIT-компиляция в контексте тензорных вычислений?
10. Какие особенности генерации кода для GPU?
11. Что такое автоматическое дифференцирование и как оно связано с компиляторами?
12. Как тензорные компиляторы взаимодействуют с фреймворками машинного обучения?
13. Какие проблемы возникают при компиляции для гетерогенных систем?
14. Какие метрики используются для оценки качества тензорного компилятора?
15. Что такое граф потока данных?
16. Как осуществляется слияние операций в тензорных компиляторах?
17. Какие методы используются для упрощения выражений?
18. Как влияет порядок измерений на производительность вычислений?
19. Что такое память с разделяемым доступом и как она учитывается при компиляции?
20. Какие технологии используются для работы с памятью ограниченного объема?
21. Какие подходы используются для снижения потребления энергии в тензорных вычислениях?
22. Что такое цикловая оптимизация в тензорных компиляторах?
23. Как происходит преобразование из высокого уровня в низкий уровень IR?
24. Какие библиотеки используются для работы с тензорами на уровне железа?
25. Какие преимущества даёт использование доменно-специфичных языков (DSL) в тензорных компиляторах?

26. Какие перспективы развития тензорных компиляторов?
27. Какие проблемы возникают при использовании JIT-компиляции в реальном времени?
28. Как тензорные компиляторы поддерживают работу с разреженными данными?
29. Какие вызовы связаны с масштабируемостью тензорных вычислений?
30. Как тензорные компиляторы адаптируются под новые типы процессоров?

Критерии оценивания (оценочное средство - Контрольные вопросы)

Оценка	Критерии оценивания
зачтено	Студент ответил на большую часть вопросов возможно с незначительными недочетами.
не зачтено	При ответе студент допускает грубые ошибки в основном материале и решении стандартных задач.

5.3.2 Типовые задания (оценочное средство - Задачи) для оценки сформированности компетенции ПК-3

1. Реализовать лексический анализатор для тензорных выражений.
2. Написать парсер для арифметических тензорных выражений.
3. Преобразовать AST в MLIR-представление

Критерии оценивания (оценочное средство - Задачи)

Оценка	Критерии оценивания
зачтено	Выполнены все или большая часть этапов решения задачи или задача решена с незначительными недочетами. Результаты работы представлены преподавателю в срок
не зачтено	Выполнены не все практические задания или выполнены не в полном объеме (представлено не полное описание этапов выполнения заданий, получен неверный ответ, результаты работы не представлены преподавателю).

6. Учебно-методическое и информационное обеспечение дисциплины (модуля)

Основная литература:

1. Малявко Александр Антонович. Формальные языки и компиляторы : учебник для вузов / А. А. Малявко. - Москва : Юрайт, 2025. - 429 с. - (Высшее образование). - ISBN 978-5-534-04288-7. - Текст : электронный // ЭБС "Юрайт"., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=925042&idb=0>.

Дополнительная литература:

1. Разработка компиляторов / Вояковская Н.Н., Москаль А.Е., Булычев Д.Ю., Терехов А.А. - Москва : ИНТУИТ, 2016., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=663586&idb=0>.

Программное обеспечение и Интернет-ресурсы (в соответствии с содержанием дисциплины):

1. Операционные системы семейства Microsoft Windows, лицензия по подписке Microsoft Imagine.
2. Браузер Google Chrome, предоставляется бесплатно на условиях лицензионных соглашений на программное обеспечение с открытым исходным кодом.
3. Среда разработки семейства Microsoft Visual Studio, лицензия по подписке Microsoft Imagine.

7. Материально-техническое обеспечение дисциплины (модуля)

Учебные аудитории для проведения учебных занятий, предусмотренных образовательной программой, оснащены мультимедийным оборудованием (проектор, экран), техническими средствами обучения, компьютерами.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ по направлению подготовки/специальности 02.03.02 - Фундаментальная информатика и информационные технологии.

Автор(ы): Золотых Николай Юрьевич, доктор физико-математических наук, доцент.

Программа одобрена на заседании методической комиссии от 02.12.2024, протокол № 5.