

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
Графические процессоры и акселераторы

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.06.02 Графические процессоры и акселераторы 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-4: Способен проектировать программное обеспечение	<p>ПК-4.1: Знает типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения</p> <p>ПК-4.2: Знает методы и средства проектирования программного обеспечения</p> <p>ПК-4.3: Знает методы и средства проектирования баз данных</p> <p>ПК-4.4: Умеет использовать существующие типовые решения и шаблоны проектирования программного обеспечения</p> <p>ПК-4.5: Умеет применять методы и средства проектирования программного обеспечения, структур данных, баз данных</p>	<p>ПК-4.1: Знать архитектуру, принципы работы графических процессоров.</p> <p>ПК-4.2: Знать условия применения графических процессоров (GPU) и аппаратных ускорителей для различных вычислений, включая обработку графики и неграфические задачи</p> <p>ПК-4.3: Знать методы и средства проектирования баз данных</p> <p>ПК-4.4: Уметь применять графические процессоры (GPU) и аппаратные ускорители для различных вычислений, включая обработку графики и неграфические задачи</p> <p>ПК-4.5: Владеть навыками, необходимыми для работы с современными вычислительными платформами, использующими GPU и другие аппаратные ускорители.</p>	<p>Тест</p> <p>Отчет по лабораторным работам</p>	<p>Зачёт:</p> <p>Контрольные вопросы</p> <p>Задачи</p>

--	--	--	--	--

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	16
- занятия семинарского типа (практические занятия / лабораторные работы)	16
- 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 архитектуру графических процессоров	12	2	2	4	8
Параллелизм и SIMD-архитектура	14	2	2	4	10
Архитектура CUDA и OpenCL	14	2	2	4	10
Устройство памяти GPU	14	2	2	4	10
Вычислительные библиотеки и фреймворки	14	2	2	4	10
Графические API и шейдеры	14	2	2	4	10
Акселераторы: FPGA и ASIC	14	2	2	4	10
Современные тенденции и перспективы	11	2	2	4	7
Аттестация	0				
КСР	1			1	
Итого	108	16	16	33	75

Содержание разделов и тем дисциплины

1. Введение в архитектуру графических процессоров
 - Общие сведения о GPU.
 - Отличие от CPU.
 - Основные производители и их особенности.
2. Параллелизм и SIMD-архитектура
 - Принципы параллельных вычислений.
 - SIMD и SIMT.
 - Потоки, блоки, ядра.
3. Архитектура CUDA и OpenCL
 - Модель программирования CUDA.
 - Базовые понятия: поток, блок, сетка.
 - Введение в OpenCL.
4. Устройство памяти GPU
 - Разделение типов памяти: глобальная, разделяемая, регистровая.
 - Особенности доступа к памяти.
 - Оптимизация использования памяти.
5. Вычислительные библиотеки и фреймворки
 - BLAS, cuDNN, Thrust.
 - Использование готовых решений.
 - Поддержка машинного обучения.
6. Графические API и шейдеры
 - OpenGL, Vulkan, DirectX.
 - Типы шейдеров.
 - Назначение и использование.
7. Акселераторы: FPGA и ASIC
 - Принципы работы FPGA.
 - Особенности ASIC.
 - Сравнение с GPU.
8. Современные тенденции и перспективы
 - Ray tracing, DLSS, AI-ускорители.
 - Перспективы развития.
 - Интеграция с облачными системами

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. Касперски К., Мулдер Э. "Архитектура компьютера. Программируемая логика. Графические процессоры" – М.: ДМК Пресс, 2021.

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-4:

1. Какой из следующих терминов относится к модели вычислений GPU?
 - a) SISD
 - +b) SIMD
 - c) MIMD
 - d) MISD
2. Какой из следующих элементов является минимальной единицей исполнения в CUDA?
 - a) Block
 - b) Grid
 - +c) Thread
 - d) Kernel
3. Какой тип памяти наиболее быстрый в GPU?
 - a) Global
 - +b) Shared
 - c) Constant
 - d) Texture

Критерии оценивания (оценочное средство - Тест)

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

5.1.2 Типовые задания (оценочное средство - Отчет по лабораторным работам) для оценки сформированности компетенции ПК-4:

Лабораторная 1. Настройка среды разработки

Цель лабораторной работы – установить и настроить среду для разработки под GPU, включая CUDA Toolkit или OpenCL SDK, и проверить корректность установки.

Время выполнения: 2 часа

Перед выполнением работы студенты получают:

- Инструкцию по установке CUDA Toolkit (для NVIDIA) или OpenCL SDK.
- Системные требования и совместимость с драйверами.
- Пример простой программы "Hello GPU" (например, вывод версии устройства).

Выполнение работы предполагает решение следующих задач:

- Установка драйверов GPU и соответствующего SDK.
- Настройка переменных окружения (PATH, LD_LIBRARY_PATH).
- Компиляция и запуск тестовой программы.
- Проверка обнаружения GPU и его характеристик (память, архитектура).
- Оформление отчета с подтверждением успешной установки и скриншотами.

Лабораторная 2. Реализация параллельного умножения матриц

Цель лабораторной работы – реализовать алгоритм умножения матриц на GPU и сравнить его производительность с CPU-реализацией.

Время выполнения: 2 часа

Перед выполнением работы студенты получают:

- Шаблон кода на CUDA C/C++ или OpenCL.
- Описание алгоритма умножения матриц ($A \times B = C$).
- Пример CPU-реализации.

Выполнение работы предполагает решение следующих задач:

- Реализация ядра (kernel) для умножения матриц.
- Настройка конфигурации потоков (grid, block).
- Выделение памяти на GPU и передача данных.
- Запуск ядра и получение результата.
- Сравнение времени выполнения с CPU-версией.
- Оформление отчета с графиками производительности.

Лабораторная 3. Использование разделяемой памяти

Цель лабораторной работы – оптимизировать доступ к памяти с использованием shared memory и измерить прирост производительности.

Время выполнения: 2 часа

Перед выполнением работы студенты получают:

- Неоптимизированный код умножения матриц.
- Описание принципа коалесцированного доступа и shared memory.

Выполнение работы предполагает решение следующих задач:

- Модификация ядра с использованием shared memory для кэширования блоков матриц.
- Реализация тайлинга (разбиения на блоки).
- Измерение времени выполнения до и после оптимизации.

- Анализ использования памяти.
- Оформление отчета с анализом ускорения.

Лабораторная 4. Работа с шейдерами

Цель лабораторной работы – создать простой шейдер с использованием OpenGL или Vulkan и отобразить графику на экране.

Время выполнения: 2 часа

Перед выполнением работы студенты получают:

- Каркас приложения на C++ с OpenGL (например, с использованием GLFW и GLAD).
- Описание структуры vertex и fragment шейдеров.

Выполнение работы предполагает решение следующих задач:

- Написание vertex шейдера для преобразования координат.
- Написание fragment шейдера для закраски.
- Компиляция и присоединение шейдеров к программе.
- Отображение простой фигуры (треугольника, квадрата).
- Визуализация результата.
- Оформление отчета с кодом и скриншотами.

Лабораторная 5. Оптимизация вычислений

Цель лабораторной работы – использовать специализированные библиотеки (cuBLAS, cuFFT) для ускорения вычислений и измерить эффективность.

Время выполнения: 2 часа

Перед выполнением работы студенты получают:

- Код, реализующий матричное умножение или преобразование Фурье вручную.
- Документацию по cuBLAS и cuFFT.

Выполнение работы предполагает решение следующих задач:

- Замена ручной реализации на вызовы cuBLAS (для матриц) или cuFFT (для FFT).
- Настройка зависимостей и линковка библиотек.
- Измерение времени выполнения.
- Сравнение с предыдущими версиями.
- Оформление отчета с выводами о преимуществах библиотечных решений.

Лабораторная 6. Работа с FPGA

Цель лабораторной работы – ознакомиться с базовыми принципами работы FPGA, настроить эмулятор и реализовать простую логику.

Время выполнения: 2 часа

Перед выполнением работы студенты получают:

- Доступ к Intel FPGA SDK for OpenCL или Xilinx Vitis.
- Описание языка описания аппаратуры (HDL) или OpenCL для FPGA.
- Пример проекта (например, светодиод мигает по таймеру).

Выполнение работы предполагает решение следующих задач:

- Установка и настройка среды разработки.
- Создание простой логики (например, сумматор).
- Синтез и загрузка на эмулятор.
- Проверка работоспособности.
- Оформление отчета с описанием архитектуры и результатов.

Лабораторная 7. Интеграция GPU и CPU

Цель лабораторной работы – реализовать обмен данными между CPU и GPU, обеспечить синхронизацию и минимизировать накладные расходы.

Время выполнения: 2 часа

Перед выполнением работы студенты получают:

- Код с асинхронными потоками выполнения.
- Описание функций `cudaMemcpyAsync`, `cudaStreamCreate`.

Выполнение работы предполагает решение следующих задач:

- Создание CUDA-потоков (streams).
- Асинхронная передача данных между CPU и GPU.
- Перекрытие вычислений и передачи данных.
- Синхронизация с использованием событий.
- Оформление отчета с анализом эффективности асинхронности.

Лабораторная 8. Исследование производительности

Цель лабораторной работы – профилировать GPU-приложение, выявить узкие места и предложить пути оптимизации.

Время выполнения: 2 часа

Перед выполнением работы студенты получают:

- Приложение с потенциальными проблемами производительности.
- Доступ к профилировщику (NVIDIA Nsight, nvprof).

Выполнение работы предполагает решение следующих задач:

- Запуск профилирования.
- Анализ загрузки GPU, использования памяти, оссирансу.
- Поиск причин низкой эффективности (например, неоптимальный доступ к памяти).
- Предложение оптимизаций.

Оформление отчета с графиками и рекомендациями

Критерии оценивания (оценочное средство - Отчет по лабораторным работам)

Оценка	Критерии оценивания
зачтено	Выполнены все или большая часть этапов решения задачи или задача решена с незначительными недочетами. Программа и результаты работы представлены преподавателю в срок.
не зачтено	Выполнены не все лабораторные работы или выполнены не в полном объеме (представлено не полное описание этапов выполнения заданий, программа работает некорректно, результаты работы не представлены преподавателю).

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 ответа	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 ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько несущественных ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Ошибок нет.	Уровень знаний в объеме, превышающем программу подготовки.
<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие	При решении стандартных задач не продемонстрированы основные умения. Имели	Продемонстрированы основные умения. Решены типовые задачи с	Продемонстрированы все основные умения. Решены все основные	Продемонстрированы все основные умения. Решены все основные	Продемонстрированы все основные умения. Решены все	Продемонстрированы все основные умения. Решены все основные задачи.

	отказа обучающегося от ответа	место грубые ошибки	негрубыми ошибками. Выполнены все задания, но не в полном объеме	задачи с негрубыми ошибками. Выполнены все задания в полном объеме, но некоторые с недочетами	задачи. Выполнены все задания в полном объеме, но некоторые с недочетами	основные задачи с отдельным и несущественными недочетами, выполнены все задания в полном объеме	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 некоторым и недочетами	Продемонстрированы базовые навыки при решении стандартных задач с некоторым и 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-4

1. Что такое GPU и чем он отличается от CPU?
2. Какова основная цель параллельных вычислений?
3. Что означает термин SIMD?
4. Чем отличаются SIMD и SIMT?
5. Что такое CUDA? Для чего она используется?
6. Что представляет собой модель вычислений CUDA?
7. Как организована память в GPU?
8. Какие типы памяти используются в CUDA/OpenCL?
9. Что такое warp/wavefront?
10. Как происходит запуск ядра в CUDA?
11. Что такое shared memory и как его использовать эффективно?
12. Как работает коалесцированный доступ к памяти?
13. Что такое шейдер и какие виды шейдеров существуют?
14. Какие графические API вы знаете?
15. Что такое OpenCL и где он применяется?
16. Какие вычислительные библиотеки используются на GPU?
17. Что такое cuDNN и как он связан с глубоким обучением?
18. Какие преимущества даёт использование FPGA?
19. Чем отличается FPGA от ASIC?
20. Что такое DLSS и как он работает?
21. Как реализуется ray tracing на современных GPU?
22. Что такое tensor core и где они применяются?
23. Как происходит взаимодействие между CPU и GPU?
24. Как профилируется производительность GPU-приложений?
25. Какие проблемы возникают при работе с памятью GPU?
26. Какие ограничения у CUDA?
27. Какие вызовы связаны с энергопотреблением GPU?
28. Какие метрики используются для оценки производительности GPU?
29. Какие перспективы развития GPU и акселераторов?
30. Как GPU используется в задачах машинного обучения?

Критерии оценивания (оценочное средство - Контрольные вопросы)

Оценка	Критерии оценивания
зачтено	Студент ответил на большую часть вопросов возможно с незначительными недочетами.

Оценка	Критерии оценивания
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-4

1. Реализовать параллельное сложение массивов на GPU.
2. Написать программу умножения матрицы на число.
3. Выполнить оптимизацию доступа к памяти.

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. Основы параллельного программирования с использованием Visual Studio 2010 / Алексеев А.А. - Москва : ИНТУИТ, 2016., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=663194&idb=0>.

Дополнительная литература:

1. Новожилов Олег Петрович. Архитектура ЭВМ и вычислительных систем : учебник для вузов / О. П. Новожилов. - 2-е изд. - Москва : Юрайт, 2025. - 505 с. - (Высшее образование). - ISBN 978-5-534-20365-3. - Текст : электронный // ЭБС "Юрайт"., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=921330&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.