

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Астраханский государственный университет имени В. Н. Татищева»
(Астраханский государственный университет им. В. Н. Татищева)

СОГЛАСОВАНО
Руководитель ОПОП

_____ М.В. Коломина

«8» сентября 2022 г.

УТВЕРЖДАЮ
Зав. кафедрой ПМИ

_____ М.В. Коломина

«8» сентября 2022 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
ПАРАЛЛЕЛЬНЫЕ ВЫЧИСЛЕНИЯ НА ГРАФИЧЕСКИХ
ПРОЦЕССОРАХ**

Составитель(-и)	Гордеев И.И., к. ф.-м. н., доцент каф. ПМИ, АГУ
Направление подготовки / специальность	01.03.02 Прикладная математика и информатика
Направленность (профиль) ОПОП	Программирование и искусственный интеллект
Квалификация (степень)	бакалавр
Форма обучения	очная
Год приема	2023
Курс	4
Семестр	7

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1.1. Целями освоения дисциплины «Параллельные вычисления на графических процессорах» являются получение представления о возможностях параллельного программирования графических процессоров для вычислений общего назначения, развитие навыков написания эффективных программ для GPU, достижение глубокого понимания устройства графического процессора.

1.2. Задачи освоения дисциплины:

- сформировать умение использовать средства разработки параллельных программ для графических процессоров;
- сформировать умение использовать средства отладки и профилировки параллельных программ для графических процессоров.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

2.1. Учебная дисциплина «Параллельные вычисления на графических процессорах» относится к части, формируемой участниками образовательных отношений (элективным дисциплинам) и осваивается в 7 семестре.

2.2. Для изучения данной учебной дисциплины (модуля) необходимы следующие знания, умения, навыки, формируемые предшествующими учебными дисциплинами (модулями):

Технологии программирования;

Параллельное программирование.

Знания: управляющие конструкции языков программирования

Умения: писать программы для решения типовых задач на обработку числовых последовательностей

Навыки: написания и отладки программ объемом 30-50 строк.

2.3. Последующие учебные дисциплины (модули) и (или) практики, для которых необходимы знания, умения, навыки, формируемые данной учебной дисциплиной (модулем):

- Проектирование программного обеспечения;
- Выпускная квалификационная работа.

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО и ОПОП ВО по данному направлению подготовки:

- УК-1. Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий.

Таблица 1 – Декомпозиция результатов обучения

Код и наименование компетенции	Планируемые результаты обучения по дисциплине (модулю)		
	Знать (1)	Уметь (2)	Владеть (3)
УК-1 УК- 1.1 Анализирует задачу, выделяя ее базовые составляющие, осуществляет декомпозицию задачи. УК- 1.2 Находит и критически анализирует информацию, необходимую для решения поставленной задачи. УК- 1.3 Рассматривает возможные, в том числе нестандартные варианты решения задачи, оценивая их достоинства и недостатки, а также возможные последствия с учетом	ИУК-1.1.1 основные принципы критического анализа; способы поиска вариантов решения поставленной проблемной ситуации.	ИУК-1.2.1 анализировать проблемную ситуацию как систему, выявляя ее составляющие и связи между ними; предлагать варианты решения поставленной проблемной ситуации на основе системного подхода.	ИУК-1.3.1 навыками критического анализа проблемных ситуаций на основе системного подхода и определения стратегии действий для достижения поставленной цели.

ценностных ориентиров.			
------------------------	--	--	--

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Объем дисциплины составляет 4 зачетные единицы, в том числе 60 часов, выделенных на контактную работу обучающихся с преподавателем (из них 60 часов – лабораторные работы), и 84 часа – на самостоятельную работу обучающихся.

Таблица 2 – Структура и содержание дисциплины (модуля)

Раздел, тема дисциплины (модуля)	Семестр	Контактная работа (в часах)			Самостоят. работа		Форма текущего контроля успеваемости, форма промежуточной аттестации
		Л	ПЗ	ЛР	КР	СР	
Тема 1. Обзор технологий программирования для графических процессоров.	7			15		21	Коллоквиум
Тема 2. Технология программирования CUDA.	7			15		21	Коллоквиум
Тема 3. Технология программирования OpenACC.	7			15		21	Коллоквиум
Тема 4. Технология программирования OpenCL.	7			15		21	Лабораторная работа.
ИТОГО				60		84	Диф. зачет (зачет с оценкой)

Примечание: Л – лекция; ПЗ – практическое занятие, семинар; ЛР – лабораторная работа; КР – курсовая работа; СР – самостоятельная работа.

Таблица 3 – Матрица соотнесения разделов, тем учебной дисциплины (модуля) и формируемых компетенций

Разделы, темы дисциплины	Кол-во часов	Код компетенции	общее количество компетенций
		УК-1	
Тема 1. Обзор технологий программирования для графических процессоров.	31	+	1
Тема 2. Технология программирования CUDA.	43	+	1
Тема 3. Технология программирования OpenACC.	27	+	1
Тема 4. Технология программирования OpenCL.	43	+	1
Итого	144		

Содержание дисциплины

Тема 1. Обзор технологий программирования для графических процессоров.

Технология программирования OpenGL. Технология программирования Vulkan (API). Технология программирования DirectX. Технология программирования C++ AMP. Технология программирования CUDA. Технология программирования OpenACC. Технология программирования OpenCL.

Тема 2. Технология программирования CUDA.

Программная модель CUDA. Иерархия памяти в CUDA. Общие методы оптимизации CUDA-программ. Стратегии повышения производительности. Максимальное использование возможностей GPU. Уровень приложения. Уровень мультипроцессора. Интерфейс расчета загруженности (Occupancy calculator). Максимизация пропускной способности памяти. Динамический параллелизм. Окружение выполнения и модель памяти. Поддержка языка C/C++. Функции языка C++ 11. Функции языка C++ 14. Программирование для унифицированной памяти. Упрощение программирования на GPU. Миграция данных и согласованность. Новое в CUDA: Volta, группы взаимодействующих нитей и другое. Поддержка новой архитектуры NVIDIA. Группы взаимодействующих нитей. Ускоренные библиотеки и новые алгоритмы.

Тема 3. Технология программирования OpenACC.

Стандарт OpenACC. Программирование GPU средствами OpenACC. Основные директивы OpenACC: kernels, parallel, data, loop. Шаги для ускорения приложения средствами OpenACC.

Тема 4. Технология программирования OpenCL.

Версии стандарта OpenCL. Используемая терминология. Память. Работа OpenCL-приложения. Определение наличия платформ и устройств. Информация о возможностях устройств. Список возможных видов информации. Контекст. Очереди команд. Работа с кодом ядра. Отдельная компиляция ядер. Язык для написания ядер. Встроенные скалярные типы данных. Встроенные векторные типы данных. Другие встроенные типы данных. Векторные литералы. Компоненты векторов. Ключевые слова. Преобразования и приведение типов. Операторы. Квалификаторы. Встроенные функции. Заголовочные файлы OpenCL. Использование C++ интерфейса. Файлы cl.hpp и cl2.hpp. Расширения (Device Extensions). Типы и имена расширений. Компиляция при использовании расширений. Информация о расширениях в файле cl_ext.h (cl_gl_ext.h). Наиболее распространённые расширения. Расширения в коде ядра — на примере cl_khr_fp64.

5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПРЕПОДАВАНИЮ И ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

5.1. Указания для преподавателей по организации и проведению учебных занятий по дисциплине (модулю)

Лабораторные занятия

Лабораторное занятие – целенаправленная форма организации педагогического процесса, направленная на углубление научно-теоретических знаний и овладение определенными методами работы, в процессе которых вырабатываются умения и навыки выполнения тех или иных учебных действий в данной сфере науки. Они развивают научное мышление и речь, позволяют проверить знания студентов и выступают как средства оперативной обратной связи.

Правильно организованные занятия ориентированы на решение следующих задач:

- обобщение, систематизация, углубление, закрепление полученных в процессе самостоятельной работы теоретических знаний по дисциплине;
- формирование практических умений и навыков, необходимых в будущей профессиональной деятельности, реализация единства интеллектуальной и практической деятельности;
- выработка при решении поставленных задач таких профессионально значимых качеств, как самостоятельность, ответственность, точность, творческая инициатива.

Самостоятельная работа

Самостоятельная работа – это вид учебной деятельности, которую студент совершает в установленное время и в установленном объеме индивидуально или в группе, без непосредственной помощи преподавателя (но при его контроле), руководствуясь сформированными ранее представлениями о порядке и правильности выполнения действий.

В учебном процессе образовательного учреждения выделяются два вида самостоятельной работы:

- 1) аудиторная – выполняется на учебных занятиях, под непосредственным руководством преподавателя и по его заданию.
- 2) внеаудиторная – выполняется по заданию преподавателя, но без его непосредственного участия. Внеаудиторные самостоятельные работы представляют собой логическое продолжение аудиторных занятий, проводятся по заданию преподавателя, который инструктирует студентов и устанавливает сроки выполнения задания.

5.2. Указания для обучающихся по освоению дисциплины (модуля)

Лабораторное занятие. Как к нему готовиться

1. Лабораторное занятие наиболее активный вид учебных занятий в вузе. Оно предполагает самостоятельную работу над учебными пособиями.

2. К каждому лабораторному занятию нужно готовиться. Подготовку следует начинать с повторения теории (по учебному пособию). После этого нужно решать задачи из предложенного домашнего задания.

Организация самостоятельной работы

1. Бюджет времени студента определяется временем, отведенным на занятия по расписанию и на самостоятельную работу. Задание и материал для самостоятельной работы дается во время учебных занятий, на этих же занятиях преподаватель осуществляет контроль за самостоятельной работой.
2. Для выполнения объема самостоятельной работы необходимо заниматься в среднем 4 часа (академических) ежедневно.
3. Начинать самостоятельные занятия следует с первых же дней семестра, установив определенный порядок, равномерный ритм на весь семестр. Полезно для этого составить расписание порядка дня.

Таблица 4 – Содержание самостоятельной работы обучающихся

Вопросы, выносимые на самостоятельное изучение	Кол-во часов	Формы работы
Тема 1. Технология программирования Vulkan (API).	21	Подготовка информационного сообщения
Тема 2. Новое в CUDA: Volta, группы взаимодействующих нитей и другое.	21	Подготовка информационного сообщения
Тема 3. Профилирование OpenACC программ.	21	Подготовка информационного сообщения
Тема 4. Использование C++ интерфейса к OpenCL. Файлы cl.hpp и cl2.hpp.	21	Подготовка информационного сообщения

5.3. Виды и формы письменных работ, предусмотренных при освоении дисциплины (модуля), выполняемые обучающимися самостоятельно

Примерные темы информационного сообщения:

1. Технология программирования Vulkan (API).
2. Технология программирования DirectX.
3. Новое в CUDA: Volta, группы взаимодействующих нитей и другое.
4. Группы взаимодействующих нитей.
5. Профилирование OpenACC программ.
6. Отладка OpenACC программ.
7. Использование C++ интерфейса к OpenCL. Файлы cl.hpp и cl2.hpp.
8. Расширения OpenCL (Device Extensions).

Требования к оформлению информационного сообщения

1. Формат страницы: A4.
2. Поля: левое - 3 см, правое - 1,5 см, верхнее - 2 см, нижнее - 2 см.
3. Текст, отформатированный с помощью стилей: Заголовок 1: шрифт - Times New Roman; размер шрифта - 16; начертание - полужирный; все буквы ПРОПИСНЫЕ; выравнивание - по центру; межстрочный интервал - полуторный; интервал после абзаца - 0,21. Заголовок 2: шрифт - Times New Roman; размер шрифта - 14; начертание - полужирный; выравнивание - по левому краю; отступ первой строки - 1,25 см; интервал перед абзацем - 0,42 см; интервал после абзаца - 0,21 см; межстрочный интервал - полуторный. Основной текст: шрифт - Times New Roman; размер шрифта - 14; межстрочный интервал - полуторный; отступ первой строки - 1,25 см; интервал после абзаца - 0,21 см; выравнивание - по ширине.
4. Создать автоматическую нумерацию глав и подглав. Разделы «Введение», «Заключение» и «Список литературы» не нумеруются.
5. Формулы должны быть набраны с помощью редактора формул.
6. Изображения, формулы, таблицы, схемы, диаграммы должны быть подписаны и пронумерованы (автоматическая подпись) с указанием ссылок на них.
7. Документ должен содержать:
 - 1) Титульный лист.

- 2) Содержание.
- 3) Основной текст
 - введение,
 - основная часть,
 - заключение.
- 4) Список использованной литературы.

ТИТУЛЬНЫЙ ЛИСТ

Титульный лист должен содержать:

1. Наименование организации, где выполнялась работа.
2. Наименование работы.
3. Сведения об авторе (должность, Ф.И.О.).
4. Место и дата выполнения работы.

СОДЕРЖАНИЕ

Содержание работы разместить на отдельном листе. Содержание должно быть сформировано автоматически и содержать все заголовки и подзаголовки с указанием номера страницы.

ОСНОВНОЙ ТЕКСТ

Введение. В аннотации (3-5 предложений) кратко указывается, какой проблеме посвящается методическая разработка, какие вопросы раскрывает, кому может быть полезна.

Основная часть. Количество разделов в основной части работы может изменяться в зависимости от объема имеющегося материала и поставленной перед собой целью. В этом разделе подробно рассматриваются все вопросы, внесенные в содержание. По ходу изложения можно представлять необходимые таблицы и рисунки. Нумерация по мере появления в тексте (например, рис. 1, таблица 3. и т. д.). Таблица должна иметь название и «шапку» с наименованием колонок.

СПИСОК ЛИТЕРАТУРЫ

В список литературы по порядку включаются те источники, которые использовались при написании работы. На все перечисленные в «Списке литературы» источники должны быть ссылки в основном тексте работы в виде номеров из списка, заключенных в квадратные скобки. Пример: [5], где 5 это номер по порядку в списке использованных источников.

6. ОБРАЗОВАТЕЛЬНЫЕ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

При реализации различных видов учебной работы по дисциплине «Параллельные вычисления на графических процессорах» могут использоваться электронное обучение и дистанционные образовательные технологии.

6.1. Образовательные технологии

Таблица 5 – Образовательные технологии, используемые при реализации учебных занятий

Раздел, тема дисциплины (модуля)	Форма учебного занятия		
	Лекция	Практическое занятие, семинар	Лабораторная работа
Тема 1. Обзор технологий программирования для графических процессоров.	Не предусмотрено	Не предусмотрено	Более глубокое изучение определенных аспектов лекционного материала на практике
Тема 2. Технология программирования CUDA.	Не предусмотрено	Не предусмотрено	Более глубокое изучение определенных аспектов лекционного материала на практике
Тема 3. Технология программирования OpenACC.	Не предусмотрено	Не предусмотрено	Более глубокое изучение определенных аспектов лекционного материала на практике
Тема 4. Технология программирования	Не	Не	Более глубокое изучение опре-

вания OpenCL.	предусмотрено	предусмотрено	деленных аспектов лекционно-го материала на практике
---------------	---------------	---------------	--

6.2. Информационные технологии

При реализации различных видов учебной и внеучебной работы используются следующие информационные технологии:

- система управления обучением LMS Moodle;
- использование возможностей Интернета в учебном процессе (рассылка заданий, предоставление выполненных работ, ответы на вопросы, ознакомление обучающихся с оценками и т.д.);
- использование электронных учебников и различных сайтов (например, электронные библиотеки, журналы и т.д.) как источник информации;
- использование возможностей электронной почты;
- использование средств представления учебной информации (электронных учебных пособий, применение новых технологий для проведения занятий с использованием презентаций и т.д.);
- использование интерактивных средств взаимодействия участников образовательного процесса (технологии дистанционного или открытого обучения в глобальной сети);
- использование интегрированных образовательных сред, где главной составляющей являются не только применяемые технологии, но и содержательная часть, т.е. информационные ресурсы (доступ к мировым информационным ресурсам, на базе которых строится учебный процесс).

6.3. Программное обеспечение, современные профессиональные базы данных и информационные справочные системы

6.3.1. Программное обеспечение

Наименование программного обеспечения	Назначение
Adobe Reader	Программа для просмотра электронных документов
Платформа дистанционного обучения LMS Moodle	Виртуальная обучающая среда
Microsoft Office 2013, Microsoft Office Project 2013, Microsoft Office Visio 2013	Пакет офисных программ
7-zip	Архиватор
Microsoft Windows 7 Professional	Операционная система
Kaspersky Endpoint Security	Средство антивирусной защиты
Google Chrome	Браузер
Microsoft Visual Studio	Среда разработки
CodeBlocks	Среда разработки
Mingw	Компилятор
Notepad++	Текстовый редактор
OpenOffice	Пакет офисных программ
WinDjView	Программа для просмотра файлов в формате DJV и DjVu
Microsoft MPI	Среда выполнения параллельных процессов
Microsoft MPI SDK	Пакет разработчика для MPI
Intel SDK for OpenCL	Пакет разработчика OpenCL

6.3.2. Современные профессиональные базы данных и информационные справочные системы

1. Электронная библиотека «Астраханский государственный университет» собственной генерации на платформе ЭБС «Электронный Читальный зал – БиблиоТех». <https://biblio.asu.edu.ru>
2. Электронно-библиотечная система (ЭБС) ООО «Политехресурс» «Консультант студента». www.studentlibrary.ru.
3. Электронная библиотечная система издательства ЮРАЙТ, раздел «Легендарные книги». www.biblio-online.ru

4. Электронный каталог Научной библиотеки АГУ на базе MARK SQL НПО «Информ-систем». <https://library.asu.edu.ru>

7. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ

7.1. Паспорт фонда оценочных средств

При проведении текущего контроля и промежуточной аттестации по дисциплине «Параллельные вычисления на графических процессорах» проверяется сформированность у обучающихся компетенций, указанных в разделе 3 настоящей программы. Этапность формирования данных компетенций в процессе освоения образовательной программы определяется последовательным освоением дисциплин и прохождением практик, а в процессе освоения дисциплины – последовательным достижением результатов освоения содержательно связанных между собой разделов, тем.

Таблица 6 – Соответствие разделов, тем дисциплины (модуля), результатов обучения по дисциплине (модулю) и оценочных средств

Контролируемый раздел, тема дисциплины (модуля)	Код контролируемой компетенции (компетенций)	Наименование оценочного средства
Тема 1. Обзор технологий программирования для графических процессоров.	УК-1	Коллоквиум
Тема 2. Технология программирования CUDA.	УК-1,	Коллоквиум
Тема 3. Технология программирования OpenACC.	УК-1	Коллоквиум
Тема 4. Технология программирования OpenCL.	УК-1	Лабораторная работа.

7.2. Описание показателей и критериев оценивания компетенций, описание шкал оценивания

Таблица 7 – Показатели оценивания результатов обучения в виде знаний

Шкала оценивания	Критерии оценивания
5 «отлично»	демонстрирует глубокое знание теоретического материала, умение обоснованно излагать свои мысли по обсуждаемым вопросам, способность полно, правильно и аргументированно отвечать на вопросы, приводить примеры
4 «хорошо»	демонстрирует знание теоретического материала, его последовательное изложение, способность приводить примеры, допускает единичные ошибки, исправляемые после замечания преподавателя
3 «удовлетворительно»	демонстрирует неполное, фрагментарное знание теоретического материала, требующее наводящих вопросов преподавателя, допускает существенные ошибки в его изложении, затрудняется в приведении примеров и формулировке выводов
2 «неудовлетворительно»	демонстрирует существенные пробелы в знании теоретического материала, не способен его изложить и ответить на наводящие вопросы преподавателя, не может привести примеры

Таблица 8 – Показатели оценивания результатов обучения в виде умений и владений

Шкала оценивания	Критерии оценивания
5 «отлично»	демонстрирует способность применять знание теоретического материала при выполнении заданий, последовательно и правильно выполняет задания, умеет обоснованно излагать свои мысли и делать необходимые выводы
4 «хорошо»	демонстрирует способность применять знание теоретического материала при выполнении заданий, последовательно и правильно выполняет задания, умеет обоснованно излагать свои мысли и делать необходимые выводы, допускает единичные ошибки, исправляемые после замечания преподавателя
3 «удовлетворительно»	демонстрирует отдельные, несистематизированные навыки, испытывает затруднения и допускает ошибки при выполнении заданий, выполняет задание по подсказке преподавателя, затрудняется в формулировке выводов

Шкала оценивания	Критерии оценивания
2 «неудовлетворительно»	не способен правильно выполнить задания

7.3. Контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности

Раздел 1. Обзор технологий программирования для графических процессоров.

Раздел 2. Технология программирования CUDA.

Раздел 3. Технология программирования OpenACC.

Вопросы к коллоквиуму

1. Преимущества вычислений на графическом процессоре.
2. Средства разработки для графического процессора.
3. Интерфейсы программирования графики и шейдерные языки.
4. Специализированные средства программирования GPU от производителей.
5. Использование технологии CUDA для вычислений.
6. Модель исполнения и иерархия потоков в CUDA.
7. Иерархия памяти в CUDA.
8. Интерфейс программирования CUDA.
9. Общие принципы вычислений на базе CUDA.
10. Директивы параллельного выполнения в OpenACC.
11. Директивы управления данными в OpenACC.

Раздел 4. Технология программирования OpenCL.

Лабораторная работа №1

Реализовать алгоритм битонной сортировки с использованием OpenCL на основе примера из Intel SDK.

Лабораторная работа №2

Реализовать метод Монте-Карло для ценообразования опционов на акции с использованием OpenCL на основе примера из Intel SDK.

Лабораторная работа №3

Реализовать алгоритм медианного фильтра с использованием OpenCL на основе примера из Intel SDK.

Лабораторная работа №4

Реализовать одновременное использование нескольких вычислительных устройств OpenCL на основе примера из Intel SDK.

Перечень вопросов и заданий, выносимых на дифференцированный зачет

1. Технология программирования OpenGL.
2. Технология программирования Vulkan (API).
3. Технология программирования DirectX.
4. Технология программирования C++ AMP.
5. Программная модель CUDA.
6. Иерархия памяти в CUDA.
7. Общие методы оптимизации CUDA-программ.
8. Стратегии повышения производительности в CUDA.
9. Уровень приложения в CUDA.
10. Уровень мультипроцессора в CUDA.
11. Программирование GPU средствами OpenACC.
12. Основные директивы OpenACC: kernels, parallel, data, loop.
13. Шаги для ускорения приложения средствами OpenACC.
14. Версии стандарта OpenCL.
15. Работа OpenCL-приложения. Определение наличия платформ и устройств.

16. Информация о возможностях устройств OpenCL. Список возможных видов информации.
17. Контекст OpenCL. Очереди команд.
18. Работа с кодом ядра OpenCL. Отдельная компиляция ядер. Язык для написания ядер.
19. Встроенные типы данных OpenCL.
20. Векторные литералы OpenCL. Компоненты векторов.

Таблица 9. Примеры оценочных средств с ключами правильных ответов

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
<i>УК-1. Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, выработать стратегию действий.</i>				
1.	Задание закрытого типа	<i>Выберите верные ответы.</i> Выберите условия реализуемости расписания параллельного алгоритма. а. на каждой вычислительной единице все операции выполняются одна за другой б. количество операций, выполняющихся на каждом вычислительном устройстве, постоянно в. вычислительные устройства, выполняющие разные операции, не могут обмениваться информацией между собой г. каждая операция выполняется не более чем на одном вычислительном устройстве	а, г	1-3
2.		<i>Выберите верный ответ.</i> Какая команда OpenGL определяет тип формируемых примитивов (точки, отрезки, треугольники и тд)? а. ClearColor б. Vertex в. Begin г. End	в	1-3
3.		<i>Выберите верные ответы.</i> Какие команды OpenGL используются для завершения формирования изображения? а. Clear б. Finish в. Vertex г. Flush	б, г	1-3
4.		<i>Выберите верный ответ.</i>	б	1-3

№ д/п	Тип задания	Формулировка задания	Правильный ответ	Время вы- полнения (в минутах)
		Каков размер матриц в OpenGL? а. 3*3 б. 4*4 в. 5*5 г. 6*6		
5.		<i>Выберите верный ответ.</i> Как располагаются элементы в матрицах OpenGL? а. горизонтально б. вертикально в. в видовой матрице горизонтально, в других матрицах - вертикально г. в видовой матрице вертикально, в других матрицах - горизонтально	б	1-3
6.	Задание открытого типа	Дайте характеристику фреймворку OpenCL.	OpenCL (англ. Open Computing Language — открытый язык вычислений) — фреймворк для написания компьютерных программ, связанных с параллельными вычислениями на различных графических и центральных процессорах, а также FPGA. В OpenCL входят язык программирования, который основан на стандарте языка программирования Си C99, и API. OpenCL обеспечивает параллелизм на уровне инструкций и на уровне данных и является осуществлением техники GPGPU. OpenCL является полностью открытым стандартом, его использование не облагается лицензионными отчислениями.	5-8
7.		Дайте характеристику программно-аппаратной архитектуре CUDA.	CUDA (изначально аббр. от англ. Compute Unified Device Architecture) — программно-аппаратная архитектура параллельных вычислений, которая позволяет существенно увеличить вычислительную производительность благодаря использованию графических процессоров фирмы Nvidia. CUDA SDK позволяет программистам реализовывать на специальных упрощённых диалектах языков программирования Си, C++ и Фортран алгоритмы, выполнимые на графических и тензорных процессорах Nvidia. Архитектура CUDA даёт разработчику возможность по своему усмотрению организовывать доступ к набору инструкций графического или тензорного ускорителя и управлять его памятью. Функции, ускоренные при помощи CUDA, можно вызывать из различных языков, в том числе Python, MATLAB и т. п.	5-8
8.		Назовите не менее трех пре-	По сравнению с традиционным подходом к	5-8

№ д/п	Тип задания	Формулировка задания	Правильный ответ	Время вы- полнения (в минутах)
		имущества CUDA.	организации вычислений общего назначения посредством возможностей графических API, у архитектуры CUDA отмечают следующие преимущества в этой области: 1.Интерфейс программирования приложений CUDA (CUDA API) основан на стандартном языке программирования Си с некоторыми ограничениями. По мнению разработчиков, это должно упростить и сгладить процесс изучения архитектуры CUDA; 2.Разделяемая между потоками память (shared memory) размером в 16 Кб может быть использована под организованный пользователем кэш с более широкой полосой пропускания, чем при выборке из обычных текстур; 3.Более эффективные транзакции между памятью центрального процессора и видеопамятью; 4.Полная аппаратная поддержка целочисленных и побитовых операций; 5.Поддержка компиляции кода GPU средствами открытого проекта LLVM.	
9.		Дайте характеристику программному стандарту OpenACC.	OpenACC (от англ. Open Accelerators) — программный стандарт для распараллеливания программ, разрабатываемый совместно компаниями Cray, CAPS, Nvidia и PGI. Стандарт описывает набор директив компилятора, предназначенных для упрощения создания гетерогенных параллельных программ, задействующих как центральный, так и графический процессоры. Как и более ранний стандарт OpenMP, OpenACC используется для аннотирования фрагментов программ на языках C, C++ и Fortran. С помощью набора директив компилятора программист отмечает участки кода, которые следует выполнять параллельно или на графическом процессоре, обозначает какие из переменных являются общими, а какие индивидуальными для потока и т. п. По синтаксису схож с OpenMP. Стандарт OpenACC позволяет программисту абстрагироваться от особенностей инициализации графического процессора, вопросов передачи данных на сопроцессор и обратно и т. д.	5-8
10		Дайте описание модели платформы OpenCL.	Платформа OpenCL состоит из хоста соединенного с устройствами, поддерживающими OpenCL. Каждое OpenCL-устройство состоит из вычислительных блоков (Compute Unit), которые далее разделяются на один или более элементы-обработчики (Processing Elements, далее PE). OpenCL-приложение выполняется на хосте в соответствии с нативными моделями его	5-8

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
			платформы. OpenCL-приложение отправляет с хоста команды устройствам на выполнение вычислений на PE. PE в рамках вычислительного блока выполняют один поток команд как SIMD блоки (одна инструкция выполняется всеми одновременно, обработка следующей инструкции не начнется, пока все PE не завершат исполнение текущей инструкции), либо как SPMD блоки (у каждого PE собственный счетчик инструкций (program counter)).	
11	Задание комбинированного типа	<i>Верно ли утверждение:</i> На эффект перспективы при использовании команды Frustum влияет значение параметра f. Ответ обоснуйте.	Утверждение неверно, поскольку на эффект перспективы при использовании команды Frustum влияет значение параметра n.	1-3

7.4. Методические материалы, определяющие процедуры оценивания результатов обучения по дисциплине

Таблица 10. Технологическая карта рейтинговых баллов по дисциплине

№ п/п	Контролируемые мероприятия	Количество мероприятий / баллы	Максимальное количество баллов	Срок представления
Основной блок				
1.	<i>Лабораторные работы</i>	4/10	40	
2.	<i>Коллоквиум</i>	3/10	30	
3.	<i>Подготовка информационного сообщения</i>	4/5	20	
Всего			90	-
Блок бонусов				
4.	<i>Посещение занятий</i>		5	
5.	<i>Своевременное выполнение всех заданий</i>		5	
Всего			10	-
ИТОГО			100	-

Таблица 11. Шкала перевода рейтинговых баллов в итоговую оценку за семестр по дисциплине

Сумма баллов	Оценка по 4-балльной шкале	
90–100	5 (отлично)	Зачтено
85–89	4 (хорошо)	
75–84		
70–74		
65–69		
60–64	3 (удовлетворительно)	
Ниже 60	2 (неудовлетворительно)	Не зачтено

При реализации дисциплины в зависимости от уровня подготовленности обучающихся могут быть использованы иные формы, методы контроля и оценочные средства, исходя из конкретной ситуации.

8. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

8.1. Основная литература

1. Гергель В.П. Теория и практика параллельных вычислений / Гергель В.П. - М.: Национальный Открытый Университет «ИНТУИТ», 2016. URL: <http://www.studentlibrary.ru/book/ISBN9785947746457.html> (ЭБС «Консультант студента»)
2. Малявко А.А., Параллельное программирование на основе технологий OpenMP, MPI, CUDA: учеб. пособие / Малявко А.А. - Новосибирск: Изд-во НГТУ, 2015. - 116 с. - ISBN 978-5-7782-2614-2 URL: <http://www.studentlibrary.ru/book/ISBN9785778226142.html> (ЭБС «Консультант студента»)
3. Тумаков Д.Н., Технология программирования CUDA : учеб. пособие / Д.Н. Тумаков, Д.Е. Чикрин, А.А. Егорчев, С.В. Голоусов - Казань : Изд-во Казан. ун-та, 2017. - 112 с. - ISBN 978-5-00019-913-8 - URL : <https://www.studentlibrary.ru/book/ISBN9785000199138.html> (ЭБС «Консультант студента»)
4. Батчер П., Семь моделей конкуренции и параллелизма за семь недель / Пол Батчер - М.: ДМК Пресс, 2015. - 360 с. - ISBN 978-5-97060-244-7 - URL: <https://www.studentlibrary.ru/book/ISBN9785970602447.html> (ЭБС «Консультант студента»)
5. Карепова Е.Д., Основы многопоточного и параллельного программирования: учеб. пособие / Карепова Е.Д. - Красноярск: СФУ, 2016. - 356 с. - ISBN 978-5-7638-3385-0 - URL: <https://www.studentlibrary.ru/book/ISBN9785763833850.html> (ЭБС «Консультант студента»)

8.2. Дополнительная литература

1. Антонов А.С. Параллельное программирование с использованием технологии MPI / Антонов А.С. - М.: Национальный Открытый Университет «ИНТУИТ», 2016. URL: http://www.studentlibrary.ru/book/intuit_240.html (ЭБС «Консультант студента»)
2. Тоуманен Б., Программирование GPU при помощи Python и CUDA / Тоуманен Б., пер. с англ. А.В. Борескова. - М.: ДМК Пресс, 2020. - 254 с. - ISBN 978-5-97060-821-0 - URL: <https://www.studentlibrary.ru/book/ISBN9785970608210.html> (ЭБС «Консультант студента»)
3. Малявко А.А., Суперкомпьютеры и системы. Мультикомпьютеры: учебное пособие / Малявко А.А. - Новосибирск: Изд-во НГТУ, 2017. - 64 с. - ISBN 978-5-7782-3294-5 - URL: <https://www.studentlibrary.ru/book/ISBN9785778232945.html> (ЭБС «Консультант студента»)
4. Уильямс, Э. Параллельное программирование на C++ в действии. Практика разработки многопоточных программ / Э. Уильямс; пер. с англ. А. А. Слинкина. - 2-е изд. - Москва : ДМК Пресс, 2023. - 674 с. Систем. требования: Adobe Reader XI либо Adobe Digital Editions 4.5 ; экран 10". - ISBN 978-5-89818-319-6. - Текст : электронный // ЭБС "Консультант студента" : [сайт]. - URL : <https://www.studentlibrary.ru/book/ISBN9785898183196.html> (ЭБС «Консультант студента»)

8.3. Интернет-ресурсы, необходимые для освоения дисциплины (модуля).

1. Электронно-библиотечная система (ЭБС) ООО «Политехресурс» «Консультант студента». www.studentlibrary.ru.

9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для проведения лабораторных занятий:

1. Лабораторные занятия проводятся с группами или подгруппами не более 15 человек.
2. Аудитория должна быть оснащена необходимым количеством столов, стульев, доской маркерной и электронной.
1. Аудитория должна иметь следующие нормы освещенности
 - СНиП 23-05-95 «Естественное и искусственное освещение» норма освещенности аудиторий ВУЗов 400 Лк.
 - СанПиН 2.2.1/2.1.1.1278-03 «Гигиенические требования к естественному, искусственному и совмещенному освещению жилых и общественных зданий» пункт 3.3.3. «Общее освещение в помещениях общественных зданий должно быть равномерным».

2. В аудитории должно быть не менее 15 компьютеров, находящихся в исправном состоянии.
3. Расположение компьютеров в аудитории должно позволять преподавателю подойти к рабочему месту студента.
4. Компьютеры должны быть соединены локальной сетью со скоростью не менее 1 Гбит/с и подключены к сети Интернет.
5. Наличие видеокамер.
6. Компьютеры должны обладать минимальными характеристиками:
 - Объем оперативной памяти 16 Гб
 - Накопитель SDD 500 Гб
 - Процессор 12th Gen Intel(R) Core(TM) i3-12100
 - Видеоадаптер Intel(R) UHD Graphics 730

Рабочая программа дисциплины (модуля) при необходимости может быть адаптирована для обучения (в том числе с применением дистанционных образовательных технологий) лиц с ограниченными возможностями здоровья, инвалидов. Для этого требуется заявление обучающихся, являющихся лицами с ограниченными возможностями здоровья, инвалидами, или их законных представителей и рекомендации психолого-медико-педагогической комиссии. Для инвалидов содержание рабочей программы дисциплины (модуля) может определяться также в соответствии с индивидуальной программой реабилитации инвалида (при наличии).