

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Б1.В.07
(индекс дисциплины)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Технологии массивно-параллельных вычислений

(наименование дисциплины)

02.03.03 Математическое обеспечение и администрирование информационных систем

Мобильные и сетевые технологии

Форма обучения: очная

Год набора: 2020

Общая трудоемкость: 6 ЗЕ

Распределение часов дисциплины по семестрам

Семестр	7	Итого
Форма контроля	КР, экзамен	
Вид занятий		
Лекции	16	16
Лабораторные	34	34
Практические		
Руководство: курсовые работы (проекты) / РГР	1	1
Промежуточная аттестация	0,35	0,35
Контактная работа	51,35	51,35
Самостоятельная работа	129	129
Контроль	35,65	35,65
Итого	216	216

Рабочую программу составил(и):

Доцент кафедры ПМИИ, к.т.н. В.С. Климов

(должность, ученое звание, степень, Фамилия И.О.)

(должность, ученое звание, степень, Фамилия И.О.)

Рецензирование рабочей программы дисциплины:



Отсутствует



Рецензент

(должность, ученое звание, степень, Фамилия И.О.)

Рабочая программа дисциплины составлена на основании ФГОС ВО и учебного плана направления подготовки

02.03.03 Математическое обеспечение и администрирование информационных систем

Срок действия рабочей программы дисциплины до «31» августа 2024 г.

УТВЕРЖДЕНО

На заседании кафедры
«Прикладная математика и информатика»

(протокол заседания № 1 от «09» сентября 2019 г.).

1. Цель освоения дисциплины

Цель освоения дисциплины – обучение разработке приложений для процессоров с массивно параллельной вычислительной архитектурой

2. Место дисциплины в структуре ОПОП ВО

Дисциплины и практики, на освоении которых базируется данная дисциплина: «Архитектура компьютеров и операционные системы», «Объектно-ориентированное программирование 1», «Объектно-ориентированное программирование 2».

Дисциплины и практики, для которых освоение данной дисциплины необходимо как предшествующее: написание выпускной квалификационной работы.

3. Планируемые результаты обучения

Формируемые и контролируемые компетенции (код и наименование)	Индикаторы достижения компетенций (код и наименование)	Планируемые результаты обучения
ПК-7 Способен понимать и применять современный математический аппарат в решении задач профессиональной деятельности	ПК-7.1 Знает современный математический аппарат в решении задач профессиональной деятельности	Знать: аспекты современного математического аппарата в решении задач профессиональной деятельности Уметь: применять современный математический аппарат в решении задач профессиональной деятельности Владеть: навыками решения задач профессиональной деятельности
	ПК-7.2 Умеет применять математический аппарат при формировании решения задачи из профессиональной деятельности	Знать: способы применения математического аппарата при формировании решения задачи из профессиональной деятельности Уметь: применять математический аппарат при формировании решения задачи из профессиональной деятельности Владеть: навыками формирования решения задачи из профессиональной деятельности
	ПК-7.3 Владеет навыками решения задач из профессиональной деятельности	Знать: способы решения задач из профессиональной деятельности Уметь: применять навыки решения задач из профессиональной деятельности Владеть: навыками решения задач из профессиональной деятельности

4. Структура и содержание дисциплины

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив, ч.	Формы текущего контроля (наименование оценочного средства)
Модуль 1. CUDA	Лекция	Тема 1. Аппаратное обеспечение GPU и шаблоны параллельной коммуникации. Примеры параллелизации последовательных алгоритмов Фундаментальные алгоритмы GPU: свертка (reduce), сканирование (scan) и гистограмма (histogram), уплотнение (compact), сегментированное сканирование (segmented scan), сортировка	7	2			
	СР	Аппаратное обеспечение GPU и шаблоны параллельной коммуникации. Примеры параллелизации последовательных алгоритмов Фундаментальные алгоритмы GPU: свертка (reduce), сканирование (scan) и гистограмма (histogram), уплотнение (compact), сегментированное сканирование (segmented scan), сортировка	7	16			
	Лаб	Лабораторная работа 1. Фундаментальные алгоритмы GPU: свертка (reduce), сканирование (scan) и гистограмма (histogram)	7	4			Отчет по лабораторной работе
	Лаб	Лабораторная работа 2. Фундаментальные алгоритмы GPU: уплотнение (compact),	7	4			Отчет по лабораторной работе

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив, ч.	Формы текущего контроля (наименование оценочного средства)
		сегментированное сканирование (segmented scan), сортировка					
	Лекция	Тема 2. Введение в CUDA: программная модель, аппаратная реализация, библиотеки CuBLAS и CuFFT, организация потока данных	7	2			
	СР	Введение в CUDA: программная модель, аппаратная реализация, библиотеки CuBLAS и CuFFT, организация потока данных	7	16			
	Лаб	Лабораторная работа 3. Библиотеки CuBLAS и CuFFTCUDA	7	4			Отчет по лабораторной работе
	Лекция	Тема 3. Модель исполнения CUDA GPU в составе компьютера. Программная модель CUDA.	7	2			
	СР	Модель исполнения CUDA GPU в составе компьютера. Программная модель CUDA.	7	16			
	Лаб	Лабораторная работа 4. Подготовка и инициализация устройства, выделение памяти и пересылка данных	7	4			Отчет по лабораторной работе
	Лекция	Тема 4. Модель памяти CUDA. Иерархия памяти: глобальная, локальная и регистровая память, разделяемая память	7	2			
	СР	Модель памяти CUDA. Иерархия памяти: глобальная, локальная и регистровая память,	7	16			

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив, ч.	Формы текущего контроля (наименование оценочного средства)
		разделяемая память					
	Лаб	Лабораторная работа 5. Модель памяти CUDA.	7	4			Отчет по лабораторной работе
	Лекция	Тема 5. Объединённая архитектура графических процессоров. Основные составные элементы аппаратной реализации GPU	7	2			
	СР	Объединённая архитектура графических процессоров. Основные составные элементы аппаратной реализации GPU	7	16			
	Лаб	Лабораторная работа 6. Аппаратная реализация GPU	7	4			Отчет по лабораторной работе
	Лекция	Тема 6. Прикладные CUDA библиотеки. Библиотека Thrust. Библиотека CUTIL. Атомарные функции. Оптимизация CUDA программ.	7	2			
	СР	Прикладные CUDA библиотеки. Библиотека Thrust. Библиотека CUTIL. Атомарные функции. Оптимизация CUDA программ.	7	16			
	Лаб	Лабораторная работа 7. Прикладные CUDA библиотеки	7	4			Отчет по лабораторной работе
Модуль 2. OpenCLиOpenACC	Лекция	Тема 7. Стандарт директивного программирования OpenACC. Модель памяти, модель исполнения OpenACC.	7	2			
	СР	Стандарт директивного программирования OpenACC. Модель	7	16			

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив, ч.	Формы текущего контроля (наименование оценочного средства)
		памяти, модель исполнения OpenACC.					
	Лаб	Лабораторная работа 8. Модель исполнения OpenACC	7	4			Отчет по лабораторной работе
	Лекция	Тема 8. Среды разработки OpenCL и GPU-утилиты. Дополнительные возможности OpenCL.	7	2			
	Лаб	CUDA, OpenCL и OpenACC	7	2			Отчет по лабораторной работе
	СР	Среды разработки OpenCL и GPU-утилиты. Дополнительные возможности OpenCL.	7	17			
	ПА	Промежуточная аттестация	7	1,35			Курсовая работа Экзамен
	Контроль			35,65			
Итого:				216			

Схема расчета итогового балла

5. Образовательные технологии

В рамках изучения дисциплины «Технологии массивно-параллельных вычислений» предусмотрено использование следующих образовательных технологий: технология традиционного обучения: лекции и лабораторные занятия, курсовая работа.

6. Методические указания по освоению дисциплины

Изучение дисциплины требует систематического и последовательного накопления знаний, следовательно, пропуски отдельных тем не позволяют глубоко освоить предмет.

В ходе лекционных занятий задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

Студент может дополнить список использованной литературы современными источниками, не представленными в списке рекомендованной литературы, и в дальнейшем использовать собственные подготовленные учебные материалы при написании курсовых и выпускных квалификационных работ.

7. Оценочные средства

7.1. Паспорт оценочных средств

Семестр	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
7	ПК-7	Комплект отчетов по лабораторным работам 1-8 Вопросы к экзамену

7.2. Типовые задания или иные материалы, необходимые для текущего контроля

7.2.1. Лабораторная работа

(наименование оценочного средства)

Типовой(ые) пример(ы) задания(ий)

Лабораторная работа 1. Фундаментальные алгоритмы GPU: свертка (reduce), сканирование (scan) и гистограмма (histogram)

Используя фундаментальные алгоритмы GPU: свертка (reduce), сканирование (scan) и гистограмма (histogram) применить алгоритм сортировки слиянием к массиву данных.

Используемое ПО (свободно распространяемое):

CUDA Toolkit от NVIDIA

IDE C++, Fortran, Python

Форма отчета по лабораторной работе

В отчет должны быть включены следующие пункты:

- титульный лист;
- цель работы;
- краткие теоретические сведения;
- описание хода выполнения работы;
- результаты выполненной работы.

Требования к оформлению

Отчёт оформляется индивидуально каждым студентом, выполнившим задания. Страницы отчёта следует пронумеровать (титульный лист не нумеруется, далее идет страница 2 и т.д.).

В отчете должны быть представлены экранные формы результатов выполнения заданий.

Объём отчёта должен быть оптимальным для понимания того, что и как сделал студент, выполняя работу. Обязательные требования к отчёту включают общую и специальную грамотность изложения, а также аккуратность оформления.

Описание работы (пример теоретической части):

Если даны две вещественные функции $f(x)$ и $g(x)$, интегрируемые на R , то с математической точки зрения свертка (reduce) представляет собой функцию вида

$$(f*g)(t) = \int_R f(\tau)g(t - \tau)d\tau$$

Пусть дана линейная система $h[n]$, которая преобразовывает единичный импульс. Эту линейную систему назовем ядром свертки.

Любой сигнал можно разложить на сумму единичных импульсов, сдвинутых во времени и умноженных на некоторый коэффициент.

Свертка данной функции с ядром h - это линейная комбинация откликов системы на входные значения $f[i]$

Операция свертки выполняется над некоторым массивом элементов и определяется оператором свертки. Оператор свертки должен быть бинарным и ассоциативным.

Операция сканирования (scan) тоже выполняется над массивом элементов, но определяется оператором сканирования и единичным элементом (identity element).

Лабораторная работа 2. Фундаментальные алгоритмы GPU: уплотнение (compact), сегментированное сканирование (segmented scan), сортировка

Используя фундаментальные алгоритмы GPU: уплотнение (compact), сегментированное сканирование (segmented scan), сортировка, применить алгоритм сортировки слиянием к массиву данных.

Используемое ПО (свободно распространяемое):

CUDA Toolkit от NVIDIA

IDE C++, Fortran, Python

Форма отчета по лабораторной работе

В отчет должны быть включены следующие пункты:

- титульный лист;
- цель работы;
- краткие теоретические сведения;
- описание хода выполнения работы;
- результаты выполненной работы.

Требования к оформлению

Отчёт оформляется индивидуально каждым студентом, выполнившим задания. Страницы отчёта следует пронумеровать (титульный лист не нумеруется, далее идет страница 2 и т.д.).

В отчете должны быть представлены экранные формы результатов выполнения заданий.

Объём отчёта должен быть оптимальным для понимания того, что и как сделал студент, выполняя работу. Обязательные требования к отчёту включают общую и специальную грамотность изложения, а также аккуратность оформления.

Лабораторная работа 3. Библиотеки CuBLASи CuFFTCUDA

Используя библиотеки CUDACuBLASи CuFFT, выполнить преобразование Фурье и операции над матрицами.

Используемое ПО (свободно распространяемое):

CUDA Toolkit от NVIDIA

IDEC++, Fortran, Python

Форма отчета по лабораторной работе

В отчет должны быть включены следующие пункты:

- титульный лист;
- цель работы;
- краткие теоретические сведения;
- описание хода выполнения работы;
- результаты выполненной работы.

Требования к оформлению

Отчёт оформляется индивидуально каждым студентом, выполнившим задания. Страницы отчёта следует пронумеровать (титульный лист не нумеруется, далее идет страница 2 и т.д.).

В отчете должны быть представлены экранные формы результатов выполнения заданий.

Объём отчёта должен быть оптимальным для понимания того, что и как сделал студент, выполняя работу. Обязательные требования к отчёту включают общую и специальную грамотность изложения, а также аккуратность оформления.

Описание работы (пример теоретической части и исполняемого кода):

CUDA Toolkit от NVIDIA включает список библиотек

- Thrust : библиотека шаблонов
- NPP: обработка изображений и сигналов
- CURAND: псевдо- и квазислучайные числа
- LIBM: стандартная библиотека математических примитивов
- CUSPARSE : линейная алгебра для разреженных матриц
- CUBLAS: линейная алгебра для плотных матриц
- CUFFT: преобразование Фурье

Особенности библиотеки CUFFT

Интерфейс подобен FFTW

- Одно-, двух- и трехмерные вещественные
- и комплексные преобразования
- Одинарная и двойная точность
- Одномерное преобразования до 128
- миллионов элементов
- Потокное асинхронное вычисление
- Ненормализованный вывод: $\text{IFFT}(\text{FFT}(A)) = \text{len}(A) * A$

Алгоритм использования CUFFT включает четыре шага:

1. Выделение памяти на GPU
2. Создание и настройка преобразования (размер, тип...)
3. Выполнение преобразования столько раз, сколько необходимо, используя указатель из шага 1
4. Уничтожение преобразования и освобождение памяти на GPU

Вызов CUFFT (фрагмент)

```
#define NX 256
#define NY 128

cufftHandle plan;
cufftComplex *idata, *odata;
cudaMalloc((void**)&idata, sizeof(cufftComplex)*NX*NY);
cudaMalloc((void**)&odata, sizeof(cufftComplex)*NX*NY);
```

Лабораторная работа 4. Подготовка и инициализация устройства, выделение памяти и пересылка данных

выделение памяти и пересылку данных с использованием библиотек CUDA.

Используемое ПО (свободно распространяемое):

CUDA Toolkit от NVIDIA

IDE C++, Fortran, Python

Форма отчета по лабораторной работе

В отчет должны быть включены следующие пункты:

- титульный лист;
- цель работы;
- краткие теоретические сведения;
- описание хода выполнения работы;
- результаты выполненной работы.

Требования к оформлению

Отчёт оформляется индивидуально каждым студентом, выполнившим задания. Страницы отчёта следует пронумеровать (титульный лист не нумеруется, далее идет страница 2 и т.д.).

В отчете должны быть представлены экранные формы результатов выполнения заданий.

Объём отчёта должен быть оптимальным для понимания того, что и как сделал студент, выполняя работу. Обязательные требования к отчёту включают общую и специальную грамотность изложения, а также аккуратность оформления.

Описание работы (пример исполняемого кода):

Инициализация устройства:

cudaGetDeviceCount, cudaSetDevice, cudaGetDevice

Выделение памяти

cudaMalloc

*if(cudaMalloc((void**)&bvCU,(size_t)neurs*sizeof(FLT32))!=cudaSuccess)*

```
throw Unsupported(this->Name, "CUDA has failed to allocate BV.");
```

```
cudaMallocPitch  
if(cudaMallocPitch((void**)&wmCU,(size_t*)&wm_pitch,ins*sizeof(FLT32),neurs  
)!=cudaSuccess) !=cudaSuccess) throw Unsupported( Unsupported(this->Name, "CUDA  
has failed to allocate CUDA has failed to allocate  
WM.");
```

Пересылка данных:

Копирование между хостом и устройством

cudaMemset

```
if(cudaMemset(ldCU,0,ns*sizeof(float))!=cudaSuccess)  
throw Unsupported(this->Name, "CUDA has failed to zero-init LD.");
```

cudaMemcpy

```
if(cudaMemcpy(bvCU,bvF32,ns*sizeof(FLT32),cudaMemcpyHostToDevice)!=cudaSucc  
ess) throw Unsupported(this->Name, "CUDA failed to upload BV.");
```

Освобождение памяти

cudaFree

```
if(cudaFree(wmCU)!=cudaSuccess)  
throw Unsupported( Unsupported(this->Name, "CUDA has failed to de CUDA has failed to  
de-allocate WM allocate WM.");
```

Лабораторная работа 5. Модель памяти CUDA.

Описать способы отладки и продемонстрировать примеры использования разделяемой памяти при операции транспонирования матрицы, в таблице отобразить влияние использования разделяемой памяти на скорость вычислений

Используемое ПО (свободно распространяемое):

CUDA Toolkit от NVIDIA

IDE C++, Fortran, Python

Форма отчета по лабораторной работе

В отчет должны быть включены следующие пункты:

- титульный лист;
- цель работы;
- краткие теоретические сведения;
- описание хода выполнения работы;
- результаты выполненной работы.

Требования к оформлению

Отчёт оформляется индивидуально каждым студентом, выполнившим задания. Страницы отчёта следует пронумеровать (титульный лист не нумеруется, далее идет страница 2 и т.д.).

В отчете должны быть представлены экранные формы результатов выполнения заданий.

Объём отчёта должен быть оптимальным для понимания того, что и как сделал студент, выполняя работу. Обязательные требования к отчёту включают общую и специальную грамотность изложения, а также аккуратность оформления.

Описание работы (пример теоретической части)

Классификация

- Системы с общей (shared) памятью
- Системы с распределённой (distributed) памятью
- Гибридные (hybrid) системы

Типы памяти в CUDA

Тип памяти	Доступ	Уровень выделения	Скорость работы
Регистры	R/W	Per-thread	Высокая(on-chip)
Локальная	R/W	Per-thread	Низкая (DRAM)
Shared	R/W	Per-block	Высокая(on-chip)
Глобальная	R/W	Per-grid	Низкая (DRAM)
Constant	R/O	Per-grid	Высокая(L1 cache)
Texture	R/O	Per-grid	Высокая(L1 cache)

- Самая быстрая – shared (on-chip) и регистры
- Самая медленная – глобальная (DRAM)
- Для ряда случаев можно использовать кэшируемую константную и текстурную память
- Доступ к памяти в CUDA идет отдельно для
 - каждой половины warp'a (half-warp) Tesla 10
 - warp'a (Tesla 20)

Лабораторная работа 6.Аппаратная реализация GPU

Произвести сложение векторов с использованием технологии CUDA

Используемое ПО (свободно распространяемое):

CUDAToolkitofNVIDIA

IDE C++, Fortran, Python

Форма отчета по лабораторной работе

В отчет должны быть включены следующие пункты:

- титульный лист;
- цель работы;
- краткие теоретические сведения;
- описание хода выполнения работы;
- результаты выполненной работы.

Требования к оформлению

Отчёт оформляется индивидуально каждым студентом, выполнившим задания. Страницы отчёта следует пронумеровать (титульный лист не нумеруется, далее идет страница 2 и т.д.).

В отчете должны быть представлены экранные формы результатов выполнения заданий.

Объём отчёта должен быть оптимальным для понимания того, что и как сделал студент, выполняя работу. Обязательные требования к отчёту включают общую и специальную грамотность изложения, а также аккуратность оформления.

Описание работы (пример теоретической части)

Аппаратная реализация:

Набор SIMD мультипроцессоров

- GPU представляет из себя набор мультипроцессоров
- Каждый мультипроцессор является SIMD-набором скалярных 32-х битных процессоров
- На каждом такте мультипроцессор исполняет одну и ту же инструкцию над группой потоков, называемой warp
- Число потоков в warp – warp size

Каждый мультипроцессор обладает:

- Файлом локальных 32-х битных регистров
- Разделяемой памятью
- Кэшем констант
- Текстурным кэшем

Каждый блок потоков состоит из warp'ов

Warp – SIMD-группа потоков фиксированного размера, состоящая из скалярных потоков с последовательными координатами.

Лабораторная работа 7. Прикладные CUDA библиотеки

Используя технологии CUDA, произвести операции над матрицами: сложение, вычитание, умножение.

Используемое ПО (свободно распространяемое):

CUDA Toolkit от NVIDIA

IDE C++, Fortran, Python

Форма отчета по лабораторной работе

В отчет должны быть включены следующие пункты:

- титульный лист;
- цель работы;
- краткие теоретические сведения;
- описание хода выполнения работы;
- результаты выполненной работы.

Требования к оформлению

Отчёт оформляется индивидуально каждым студентом, выполнившим задания. Страницы отчёта следует пронумеровать (титульный лист не нумеруется, далее идет страница 2 и т.д.).

В отчете должны быть представлены экранные формы результатов выполнения заданий.

Объём отчёта должен быть оптимальным для понимания того, что и как сделал студент, выполняя работу. Обязательные требования к отчёту включают общую и специальную грамотность изложения, а также аккуратность оформления.

Описание работы (пример теоретической части и исполняемого кода)

Библиотека thrust в CUDA

Thrust - это библиотека, основанная на использовании шаблонов (template) языка C++. Все классы и функции в этой библиотеке – шаблонные.

Библиотека предоставляет в распоряжение разработчика набор различных параллельных примитивов, таких как различные преобразования, сортировка, операции

reduce и scan. Применяя thrust, многие действия могут быть записаны с использованием минимального объема кода. Все последние версии CUDA включают в себя thrust, так что для работы с thrust никаких дополнительных установок не понадобится.

- нет необходимости работать с ядрами GPU;
- STL-подобный дизайн (контейнеры/итераторы/алгоритмы);
- прозрачные операции с данными в памяти GPU;
- набор утилитных функций и алгоритмов;
- поставляется с CUDA SDK 4.0.

Функтор для генерации случайного числа в диапазоне 0..10

```
class random {  
public:  
    int operator() () {  
return rand() % 10;  
    }  
};
```

Подсчет суммы квадратов элементов на GPU

```
int final_sum = thrust::transform_reduce(gpudata.begin(), gpudata.end(),  
square<int>(), 0, thrust::plus<int>());
```

Лабораторная работа 8. Модель исполнения OpenACC

Реализовать алгоритм построения дерева принятия решений ID3 с использованием технологии CUDA.

Используемое ПО (свободно распространяемое):

CUDA Toolkit от NVIDIA

IDE C++, Fortran, Python

Форма отчета по лабораторной работе

В отчет должны быть включены следующие пункты:

- титульный лист;
- цель работы;
- краткие теоретические сведения;
- описание хода выполнения работы;
- результаты выполненной работы.

Требования к оформлению

Отчёт оформляется индивидуально каждым студентом, выполнившим задания. Страницы отчёта следует пронумеровать (титульный лист не нумеруется, далее идет страница 2 и т.д.).

В отчете должны быть представлены экранные формы результатов выполнения заданий.

Объём отчёта должен быть оптимальным для понимания того, что и как сделал студент, выполняя работу. Обязательные требования к отчёту включают общую и специальную грамотность изложения, а также аккуратность оформления.

Описание работы (пример исполняемого кода)

Пример SAXPY на C: OpenACC

```
void saxpy(int n, float a, float *x,
```

```
float *restrict y){
#pragma acc parallel
for (int i = 0; i < n; ++i)
y[i] = a*x[i] + y[i];
}
```

```
...
saxpy(1<<20, 2.0, x, y);
...
```

OpenACC API

Директивы компилятору указывают области параллельных вычислений в языках C и Fortran

- выгружает области с интенсивными вычислениями на GPU.
- код не зависит от ОС, CPU, ускорителя, и компилятора

Возможность создавать высокоуровневые гибридные (CPU+ускоритель) программы

- без явной инициализации ускорителя
- без явного копирования данных между CPU и ускорителем

Программная модель позволяет программисту легко начать программировать GPU, обеспечивая компилятор подсказками:

- данные для размещения на ускорителе и их характеристики
- рекомендации по отображению циклов на ускоритель
- и другие, связанные с производительностью, детали

Совместим с другими языками программирования GPU и библиотеками:

- взаимодействие с CUDA C/Fortran и GPU библиотеками
- например CUFFT, CUBLAS, CUSPARSE, и т.д.

7.3.2. Критерии и нормы оценки

Семестр	Форма проведения промежуточной аттестации	Критерии и нормы оценки	
7	Отчет по лабораторной работе	«зачтено»	Отметка «зачтено» ставится студенту, который продемонстрировал результаты выполнения лабораторной работы, соответствующие поставленным задачам, и предоставил отчет, оформленный должным образом и содержащий краткое описание полученных результатов
		«не зачтено»	Отметка «не зачтено» ставится студенту, который не продемонстрировал результаты выполнения лабораторной работы или не представил по ней отчет или представленный отчет не соответствует требованиям по оформлению.

Выполнение курсовой работы

Темы курсовых работ

№ п/п	Темы
1	Реализация обучения нейронной сети с использованием технологии CUDA
2	Реализация алгоритма построения дерева принятия решений с использованием алгоритма ID3 и технологии CUDA
3	Реализация алгоритма построения дерева принятия решений с использованием алгоритма C4.5 и технологии CUDA
4	Реализация алгоритма построения дерева принятия решений с использованием алгоритма CART и технологии CUDA
5	Реализация алгоритма kNN с использованием технологии CUDA
6	Реализация алгоритма kmeans с использованием технологии CUDA
7	Реализация алгоритма построения дерева принятия решений с использованием алгоритма Apriori и технологии CUDA
8	Реализация генетического алгоритма с использованием технологии CUDA
9	Реализация анализа изображения с использованием метода Viola-Jones и технологии CUDA
10	Реализация анализа изображения с использованием метода HOG SVM и технологии CUDA
11	Реализация алгоритма поиска усеченных деревьев CART и технологии CUDA
12	Реализация обучения нейро-нечеткой сети с использованием технологии CUDA
13	Реализация модели генитора с использованием технологии CUDA
14	Поиск суммы всех элементов дерева с использованием технологии CUDA
15	Умножения матрицы на вектор с использованием технологии CUDA
16	Разработка алгоритма по задаче «Обедающие философы» с использованием технологии CUDA
17	Нахождения обратной матрицы с использованием технологии CUDA
18	Скалярное произведение векторов с использованием технологии CUDA
19	Решения СЛАУ методом простых итераций с использованием технологии CUDA
20	Сложение векторов с использованием технологии CUDA
21	Нахождения определителя матрицы с использованием технологии CUDA
22	Решение системы линейных алгебраических уравнений прямым методом Гаусса с использованием технологии CUDA
23	Поиск подграфа в неориентированном графе с использованием технологии CUDA
24	Поиск циклов в графе с использованием технологии CUDA
25	Численное дифференцирование с использованием технологии CUDA
26	Вычисление интеграла методом Монте-Карло с использованием технологии CUDA.
27	Численное интегрирование с использованием технологии CUDA.
28	Решение дифференциальных уравнений с использованием технологии CUDA
29	Изучение спектров случайных процессов с использованием технологии CUDA
30	Решения СЛАУ итерационными методами вариационного типа с использованием технологии CUDA.
31	Определение собственных чисел и собственных векторов с использованием технологии CUDA.
32	Итерационный алгоритм численного интегрирования с использованием технологии CUDA.
33	Пирамидальная сортировка с использованием технологии CUDA.
34	Выполнение операций над красно-черными деревьями с использованием технологии CUDA.

№ п/п	Темы
35	Выполнение операций над биномиальной кучей с использованием технологии CUDA.
36	Выполнение операций над фибоначчиевой кучей с использованием технологии CUDA.
37	Поиск в ширину в графе с использованием технологии CUDA
38	Поиск в глубину в графе с использованием технологии CUDA
39	Очереди с приоритетами (реализации на массивах и списках, сортирующее дерево, биномиальная очередь) с использованием технологии CUDA
40	Работа со списками пропусков с использованием технологии CUDA
41	Выполнение операций над B-деревьями с использованием технологии CUDA
42	Представление линейного списка в виде сбалансированных деревьев с использованием технологии CUDA
43	Реализация функций динамического распределения памяти с использованием технологии CUDA
44	Параллельные распределители памяти с использованием технологии CUDA.
45	Выполнение операций над Trie-деревом с использованием технологии CUDA
46	Выполнение операций над Patricia-деревом с использованием технологии CUDA

Краткое описание и регламент выполнения

Написание курсовой работы требует демонстрации навыков применения теоретических и практических знаний, накопленных во время лекционной и лабораторной работы студента.

Студент может дополнить список использованной литературы современными источниками, не представленными в списке рекомендованной литературы, и в дальнейшем использовать собственные подготовленные учебные материалы.

А также использовать любое доступное ПО, реализующие задачи программирования графических процессоров.

Критерии оценки:

«отлично» - студент должен продемонстрировать результаты выполнения курсовой работы, соответствующие поставленным задачам, и предоставить отчет, оформленный должным образом и содержащий описание полученных результатов

«хорошо» - студент должен продемонстрировать результаты выполнения курсовой работы, соответствующие поставленным задачам, и предоставить отчет, содержащий описание полученных результатов

«удовлетворительно» - студент должен продемонстрировать результаты выполнения курсовой работы, соответствующие поставленным задачам, и предоставить отчет

«неудовлетворительно» - студент не продемонстрировал результаты выполнения курсовой работы

7.3.Оценочные средства для промежуточной аттестации по итогам освоения дисциплины

7.3.1. Вопросы к промежуточной аттестации

Семестр 7

№ п/п	Вопросы к экзамену
1	Что такое кластер в контексте параллельного программирования?
2	Какова асинхронная модель параллельных вычислений?
3	Какова синхронная модель параллельных вычислений?
4	Что представляют независимые операции, операции, выполняемые параллельно?
5	Каковы конвейеризация и суперскалярность?
6	Что собой представляют состояние и свойства параллельной программы?
7	Какова технология MPI в параллельном программировании?
8	Как происходит обмен данными между разными MPI-программами?
9	Каково количество процессов в MPI программе?
10	Каковы модели параллельных вычислений?
11	Что такое ускорение параллельной программы?
12	Что такое эффективность параллельной программы?
13	Какова стоимость вычислений по параллельной программе?
14	Что такое сверхлинейное ускорение?
15	Каковы виды декомпозиции при разработке параллельных программ?
16	Каковы алгоритмы сортировки в параллельном программировании?
17	Почему интерфейс OpenMP задуман как стандарт параллельного программирования?
18	Каковы принципы организации параллелизма в OpenMP?
19	Каковы типы директив Open MP?
20	Что такое директива for для распределения вычислений в параллельной области ?
21	Что такое директива sections для распределения вычислений в параллельной области?
22	Что такое директива single для распределения вычислений в параллельной области?
23	Что такое директива barrier в Open MP?
24	Что такое директива atomic в Open MP?
25	Что такое директива flush в Open MP?
26	Что такое взаимное исключение в Open MP ?
27	Что такое переменные окружения в Open MP ?
28	Что собой представляет аппаратное обеспечение GPU?
29	Каковы шаблоны параллельной коммуникации?
30	Каковы фундаментальные алгоритмы GPU?
31	Что такое библиотеки CuBLAS и CuFFT CUDA?
32	Какова программная модель CUDA?
33	Какова модель исполнения CUDA?
34	Какова модель памяти CUDA?
35	Какова архитектура графических процессоров?
36	Каковы основные составные элементы аппаратной реализации GPU?
37	Каковы прикладные CUDA библиотеки?
38	Что такое библиотека Thrust CUDA?
39	Что такое библиотека CUTIL CUDA?
40	Каковы атомарные функции CUDA?
41	Какова оптимизация CUDA программ?
42	Каков стандарт директивного программирования OpenACC?

№ п/п	Вопросы к экзамену
43	Каковы модель памяти, модель исполнения OpenACC?
44	Каковы среды разработки OpenCL и GPU-утилиты ?
45	Каковы дополнительные возможности OpenCL?
46	Каково математическое описание алгоритма GPU свертка (reduce)?
47	Каково математическое описание алгоритма GPU сканирование (scan)?
48	Каково математическое описание алгоритма GPU гистограмма (histogram)?
49	Каково математическое описание алгоритма GPU уплотнение (compact)?
50	Каково математическое описание алгоритма GPU сегментированное сканирование (segmented scan)?
51	Каковы основные математические операции поддерживаемые библиотекой CuBLAS?
52	Каковы основные математические операции поддерживаемые библиотекой CuFFT?
53	Каковы основные математические операции поддерживаемые библиотекой Thrust?
54	Каковы основные математические операции поддерживаемые библиотекой CUTIL?
55	В решении каких задач профессиональной деятельности эффективно использование библиотеки CuBLAS?
56	В решении каких задач профессиональной деятельности эффективно использование библиотеки CuFFT?
57	В решении каких задач профессиональной деятельности эффективно использование библиотеки Thrust?
58	В решении каких задач профессиональной деятельности эффективно использование библиотеки CUTIL?
59	В решении каких задач профессиональной деятельности эффективно использование алгоритма GPU сегментированное сканирование (segmented scan)?
60	В решении каких задач профессиональной деятельности эффективно использование алгоритма GPU уплотнение (compact)?

7.3.2. Критерии и нормы оценки

Семестр	Форма проведения промежуточной аттестации	Критерии и нормы оценки	
7	Экзамен	«отлично»	Студент должен знать определения, методы и технологии программирования графических процессоров (GPU) NVIDIA
		«хорошо»	Студент должен разбираться в основных терминах и понятиях. Знает основные методы и технологии программирования графических процессоров (GPU) NVIDIA
		«удовлетворительно»	Студент должен разбираться в основных терминах и понятиях. Знает некоторые методы и технологии программирования графических процессоров (GPU) NVIDIA
		«неудовлетворительно»	Студент не знает наиболее важные определения, не может ориентироваться в материале

8. Учебно-методическое и информационное обеспечение дисциплины

8.1. Обязательная литература

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно-методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
1	Р. Роби, Д. Замора	Роби, Р. Параллельные и высокопроизводительные вычисления / Р. Роби, Д. Замора ; перевод с английского А. В. Логунова. — Москва : ДМК Пресс, 2022. — 800 с. — ISBN 978-5-97060-936-1.	учебное пособие	2022	ЭБС «Лань» Режим доступа : https://e.lanbook.com/book/241124
2	Д. В. Парфенов, Д. А. Петрусевич	Парфенов, Д. В. Параллельные и распределенные вычисления : учебное пособие / Д. В. Парфенов, Д. А. Петрусевич. — Москва : РТУ МИРЭА, 2022. — 92 с.	учебное пособие	2022	ЭБС «Лань» Режим доступа https://e.lanbook.com/book/265658
3	Гергель В.П.	Теория и практика параллельных вычислений [Электронный ресурс]	учебное пособие	2020	Режим доступа: http://www.iprbookshop.ru/89478.html .— ЭБС «IPRbooks»
4	Богачёв К.Ю.	Основы параллельного программирования [Электронный ресурс]	учебное пособие	2020	Режим доступа: http://www.iprbookshop.ru/20702.html .— ЭБС «IPRbooks»

8.2. Дополнительная литература

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно-методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
1	Соснин В.В., Балакшин П.В.	Введение в параллельные вычисления [Электронный ресурс]	учебное пособие	2015	Режим доступа: http://www.iprbookshop.ru/68646.html .— ЭБС «IPRbooks»
2	Боресков А.В. [и др.].	Параллельные вычисления на GPU. Архитектура и программная модель CUDA [Электронный ресурс]	учебное пособие	2015	Режим доступа: http://www.iprbookshop.ru/54647.html .— ЭБС «IPRbooks»
3	Некрасов К.А. [и др.].	Параллельные вычисления общего назначения на графических процессорах [Электронный ресурс]	учебное пособие	2016	Режим доступа: http://www.iprbookshop.ru/69657.html .— ЭБС «IPRbooks»

8.3. Перечень профессиональных баз данных и информационных справочных систем

- 1) ИНТУИТ. Национальный открытый университет [Электронный ресурс]. – Режим доступа: <http://www.intuit.ru/>. – Загл. с экрана.
- 2) Научная электронная библиотека eLIBRARY.RU [Электронный ресурс]. – Режим доступа: <http://elibrary.ru/defaultx.asp>. – Загл с экрана.
- 3) Открытое образование [Электронный ресурс]. – Режим доступа: <https://openedu.ru/>. – Загл с экрана.

8.4. Перечень программного обеспечения

№ п/п	Наименование ПО	Реквизиты договора (дата, номер, срок действия)
1	Windows: WinPro 10 RUS Upgrd OLP NL Acdmc	договор № 757 от 04.07.2018, срок действия – бессрочно; контракт № 1653 от 14.12.2018, срок действия – бессрочно
2	Office Standard: Office Stdandard 2013 Russian OLP NL AcademicEdition Office Stdandard 2016 Russian OLP NL AcademicEdition Office Stdandard 2016 Russian OLP NL AcademicEdition	контракт № 690 от 19.05.2015, срок действия – бессрочно договор № 757 от 04.07.2018, срок действия – бессрочно контракт № 727 от 20.07.2016, срок действия – бессрочно
3	CUDA Toolkit от NVIDIA	Свободно распространяемое ПО
4	IDE C++, Fortran, Python	Свободно распространяемое ПО
5	Mathcad Education - University Edition Subscription (25 pack)	контракт № 469 от 05.06.2020, срок действия – бессрочно

8.5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

№ п/п	Наименование оборудованных учебных кабинетов, лабораторий, мастерских и др. объектов для проведения практических и лабораторных занятий, помещений для самостоятельной работы обучающихся (номер аудитории)	Перечень основного оборудования
1.	Компьютерный класс. Учебная аудитория для проведения занятий лекционного типа. Учебная аудитория для проведения занятий семинарского типа. Учебная аудитория для проведения лабораторных работ. Учебная аудитория для курсового проектирования (выполнения курсовых работ). Учебная аудитория для проведения групповых и индивидуальных	Компьютер (монитор 19", системный блок Pentium (R) Dual-Core E5500 2,8 GHz / 4 Gb / 500 Gb), столы ученические, столы компьютерные, стол преподавательский, стулья, доска аудиторная(меловая)

№ п/п	Наименование оборудованных учебных кабинетов, лабораторий, мастерских и др. объектов для проведения практических и лабораторных занятий, помещений для самостоятельной работы обучающихся (номер аудитории)	Перечень основного оборудования
	консультаций Учебная аудитория для проведения занятий текущего контроля и промежуточной аттестации., УЛК-401	
2.	Учебная аудитория для проведения занятий лекционного типа. Учебная аудитория для проведения занятий семинарского типа. Учебная аудитория для курсового проектирования (выполнения курсовых работ). Учебная аудитория для проведения групповых и индивидуальных консультаций. Учебная аудитория для проведения занятий текущего контроля и промежуточной аттестации.УЛК-418	Столы ученические двухместные (моноблок), доска аудиторная 3-х секционная (меловая), стол преподавательский , стулья, проектор Acer
3.	Помещение для самостоятельной работы студентов (Г-401)	Столы, стулья, компьютеры
4.	Помещение для самостоятельной работы обучающихся (С-401).	Шкафы для документации, доски магнитные, столы письменные, столы компьютерные