Федеральное государственное бюджетное образовательное учреждение высшего образования
Московский государственный университет имени М.В. Ломоносова
ФИЛИАЛ МГУ В Г. ДУБНЕ

УТВЕРЖДАЮ

И.о. директора

филиала МГУ в г. Дубне

_____/ Э.Э. Боос /

«01» сентября 2025г.

Филиал Московского государственного Университета имени М.В. Ломоносова в городе Дубне

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Наименование дисциплины:

Вычислительные экзафлопсные технологии (на английском языке)

Уровень высшего образования:

Магистратура

Направление подготовки / специальность:

01.04.02 «Прикладная математика и информатика»

Направленность (профиль) ОПОП:

Методы и технологии обработки данных в гетерогенных вычислительных средах

Форма обучения:

очная

Дубна 2025

Рабочая программа дисциплины (модуля) разработана в соответствии с самостоятельно установленным МГУ образовательным стандартом (ОС МГУ) для реализуемых основных профессиональных образовательных программ высшего образования по направлению подготовки 01.04.02 "Прикладная математика и информатика" программы магистратуры в редакции приказа МГУ №_____ от_____20__ г.

1. МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ) В СТРУКТУРЕ ОПОП ВО:

Дисциплина относится к вариативной части магистерской программы ОПОП ВО.

2. Входные требования для освоения дисциплины (модуля), предварительные условия (если есть):
Учащиеся должны владеть знаниями по операционным системам, компьютерным сетям, базам данных, дискретной математике и основам кибернетики.

3. Результаты обучения по дисциплине (модулю), соотнесенные с требуемыми компетенциями выпускников.

Планируемые результаты обучения по дисциплине (модулю)		
Содержание и код компетенции.	**Индикатор (показатель) достижения компетенции**	**Планируемые результаты обучения по дисциплине, сопряженные с индикаторами достижения компетенций**
СПК-2. Способность понимать и применять в исследовательской и прикладной деятельности современные суперкомпьютерные технологии, математический аппарат, вычислительные методы для проведения крупномасштабного математического моделирования и обработки данных на современных высокопроизводительных вычислительных системах	СПК-2.1. Применяет в исследовательской и прикладной деятельности современные суперкомпьютерные технологии, математический аппарат, вычислительные методы для проведения крупномасштабного математического моделирования и обработки данных на современных высокопроизводительных вычислительных системах	– современные проблемы математики, вычислительной математики; – постановку проблем физико-математического моделирования; – о взаимосвязях и фундаментальном единстве естественных наук. У1 Уметь – формализовать теоретическую проблему, найти способ и алгоритм её решения; – эффективно использовать на практике теоретические компоненты науки: понятия, суждения, умозаключения, законы; – представить панораму универсальных методов и законов современного естествознания; – работать на современных вычислительных комплексах; – абстрагироваться от несущественных влияний при моделировании

		реальных физических ситуаций; – планировать оптимальное проведение расчета. В1 Владеть – научной картиной мира; – математическим моделированием физических задач; - навыками самостоятельной работы на современных вычислительных комплексах.

4. Объем дисциплины (модуля) составляет 4 з.е., в том числе 54 академических часов, отведенных на контактную работу обучающихся с преподавателем, 90 академических часов на самостоятельную работу обучающихся.

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ), СТРУКТУРИРОВАННОЕ ПО ТЕМАМ (РАЗДЕЛАМ) С УКАЗАНИЕМ ОТВЕДЕННОГО НА НИХ КОЛИЧЕСТВА АКАДЕМИЧЕСКИХ ЧАСОВ И ВИДЫ УЧЕБНЫХ ЗАНЯТИЙ

5.1. Структура дисциплины (модуля) по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий (в строгом соответствии с учебным планом)

Наименование разделов и тем дисциплины (модуля), **Форма промежуточной аттестации по дисциплине (модулю)**	**Номинальные трудозатраты обучающегося**			**Всего академических часов**	**Форма текущего контроля успеваемости* (наименование)**
	Контактная работа (работа во взаимодействии с преподавателем) Виды контактной работы, академические часы		**Самостоятельная работа обучающегося, академические часы**		
	Занятия лекционного типа	Занятия семинарского типа			
Тема 1. Естественные вычислительные модели	2	0	6	8	решение задач, тестирование
Тема 2. Методы Монте-Карло	2	0	6	8	решение задач, тестирование
Тема 3. Клеточные	2	0	6	8	решение

автоматы					задач, тестирование
Тема 4. Системы Линденмайера	2	0	6	8	решение задач, тестирование
Тема 5. Марковские системы	2	0	6	8	решение задач, тестирование
Тема 6. Сети Петри	2	0	6	8	решение задач, тестирование
Тема 7. Нейронные сети	2	2	6	10	решение задач, тестирование
Тема 8. Генетические алгоритмы	2	2	6	10	решение задач, тестирование
Тема 9. Муравьиные алгоритмы	2	2	6	10	решение задач, тестирование
Тема 10. Модели роевого интеллекта	2	2	6	10	решение задач, тестирование
Тема 11. Метод роя частиц	2	2	6	10	решение задач, тестирование
Тема 12. Алгоритм бактериального поиска	2	2	6	10	решение задач, тестирование
Тема 13. Алгоритмы роевой робототехники	4	2	6	12	решение задач, тестирование
Тема 14. Мембранные системы	4	2	6	12	решение задач, тестирование
Тема 15. ДНК-вычисления	4	2	6	12	решение задач, тестирование
Промежуточная аттестация (экзамен)					экзамен
Итого	36	18	90	144	—

5.2. Содержание разделов (тем) дисциплины

№ п/п	Наименование разделов (тем) дисциплины	Содержание разделов (тем) дисциплин
1.	Тема 1. Современные суперкомпьютерные архитектуры	Виды параллелизма - MIMD, SIMD, потоковая обработка. Гибридная кластерная система и ее характеристики. Виды вычислительных устройств и систем. Топологии вычислительных систем. Уровни параллелизма и средства разработки. Параллелизм CPU ядра. Архитектура вычислительного узла. Принципы работы

		оперативной памяти. Свойства каналов передачи данных. Оценка времени передачи данных между вычислительными узлами. Области применения многопроцессорных систем. Суперкомпьютерное моделирование сплошной среды. Примеры газодинамических расчетов.
2.	Тема 2. Параллельные алгоритмы, основные понятия и принципы построения	Внутренний параллелизм. Оценка сложности, ускорения, эффективности, масштабируемости параллельного алгоритма. Оценка времени выполнения программы. Закон Амдаля. Сверхлинейное ускорение. Статическая и динамическая балансировка загрузки процессоров. Методы: сдваивание, геометрический параллелизм, конвейерный параллелизм, коллективное решение, диффузная балансировка. Статическая и динамическая балансировка загрузки при решении задач, использующих нерегулярные расчетные сетки.
3.	Тема 3. Технологии распараллеливания для MIMD систем с общей и распределенной памятью, организация взаимодействия параллельных процессов	Виды обменов данными (MPI). Организация обмена данными, оптимизация. Сокрытие обменов за вычислениями. Процессы и потоки в операционных системах. Потоки и семафоры стандарта Posix. Критические секции и блокировка потоков. Синхронизация процессов. Основные директивы OpenMP, простейшие примеры. Накладные расходы и проблемы. Способы устранения конфликтов по доступу к данным на примере сеточного метода.
4.	Тема 4. Гетерогенные вычисления на процессорах и ускорителях	Устройство графического процессора GPU. Потоковая обработка. Вычислительный стандарт OpenCL. Организация доступа к памяти. Примеры реализации вычислений. Методика портирования kernel-кода. Многоуровневая декомпозиция. Сокрытие обменов за вычислениями на ускорителях. Примеры реализации вычислений.
5.	Тема 5. Декомпозиция сеточных графов	Примеры расчетных сеток. Критерии декомпозиции графов. Критерий 1: классический критерий декомпозиции графа. Критерий 2: выделение обособленных доменов. Критерий 3: минимизация максимальной степени домена. Критерий 4: обеспечение связности графов каждого из доменов. Декомпозиция на основе исходной нумерации узлов. Рекурсивная бисекция. Декомпозиция регулярных графов. Методы декомпозиции произвольных графов. Иерархическая декомпозиция. Координатная бисекция. Спектральная бисекция. Алгоритм инкрементного роста доменов. Декомпозиция больших сеток. Двухуровневая стратегия обработки и хранения сеток. Пакеты декомпозиции графов.
6.	Тема 6. Обработка и визуализация данных, заданных на решетках	Клиент-серверная технология. Online или Offline визуализация: плюсы и минусы. Этапы визуализации. Визуализация изоповерхностей.

		Аппроксимация изоповерхности. Способы описания триангуляции. Метод редукции. Параллельные алгоритмы построения аппроксимирующих триангуляций. Многоуровневое огрубление больших сеток. Примеры визуализации. Ввод-вывод сеточных данных. Соотношение времени чтения данных и времени их обработки Распределенный ввод-вывод. Огрубление и сжатие скалярных сеточных функций: без потери точности, с контролируемой потерей точности. Форматы хранения данных. Пакеты визуализации данных большого объёма.
7.	Тема 7. Отказоустойчивые алгоритмы для высокопроизводительных систем	Контрольные точки. Локальные контрольные точки. Принцип локальности для алгоритмов на основе явных разностных схем. Функции MPI, обеспечивающие возможность работы в условиях отказа части вычислительных узлов. Эффективность отказоустойчивых алгоритмов. Распределенное хранение локальных контрольных точек. Совмещение во времени этапов распределенной записи локальных контрольных точек и вычислений.
8.	Тема 8. Параллельные алгоритмы решения прикладных задач	Параллельные алгоритмы сортировки данных. Алгоритмы генерации псевдослучайных чисел для суперкомпьютерных приложений. Балансировка загрузки процессоров при решении задач численного интегрирования, многомерной оптимизации, решении систем линейных алгоритмов. Динамическая балансировка загрузки процессоров при моделировании процессов горения.

6. Фонд оценочных средств (ФОС, оценочные и методические материалы) для оценивания результатов обучения по дисциплине (модулю).

6.1. Типовые контрольные задания или иные материалы для проведения текущего контроля успеваемости

Примеры заданий для самостоятельной работы студентов

Практическое самостоятельное задание № 1

Многопоточная реализация солвера CG для СЛАУ с
разреженной матрицей, заданной в формате ELLPACK.
Требования к отчету:
1. Титульный лист, содержащий
1.1 Название курса
1.2 Название задания
1.3 Фамилию, Имя, Отчество(при
наличии)
1.4 Номер группы
1.5 Дата подачи
Содержание отчета:
2 Описание задания и программной
реализации
2.1 Краткое описание задания
2.2 Краткое описание программной
реализации как организованы данные,
какие функции реализованы (название,
аргументы, назначение)
2.2.1 Указать как программа запускается (ставится в очередь) с какими
параметрами, с описанием этих параметров
2.3 Описание опробованных способов оптимизации последовательных
вычислений (по желанию)
3 Исследование производительности
3.1 Характеристики вычислительной системы:
описание одной или нескольких систем, на которых выполнено
исследование (подойдет любой многоядерный процессор). тип
процессора, количество ядер, пиковая производительность, пиковая
пропускная способность памяти. по желанию - промерять и на своем
декстопе/ноуте, и на кластере
3.1.1 Указать здесь и или в следующих пунктах как программа
компилировалась (каким компилятором, с какими параметрами)
3.2 Результаты измерений производительности
3.2.1 Последовательная производительность
Для каждой из трех базовых операций и для всего алгоритма солвера
исследовать зависимость достигаемой производительности от размера
системы N, построить графики GFLOPS от N. Нескольких N
достаточно: N=1000, 10000, 100000, 1000000
Для повышения точности измерений, замеры времени лучше производить, выполняя набор операций многократно в цикле, чтобы осреднить время измерений. Суммарное время измерений чтобы получалось порядка нескольких секунд.Оценить выигрыш от примененной оптимизации (по желанию)
3.2.2. Параллельное ускорение
Измерить OpenMP ускорение для различных N для каждой из 3-х базовых операций и для всего алгоритма солвера. При фиксированном числе N варьируется число нитей и измеряется параллельное ускорение.
4. Анализ полученных результатов
4.1 Процент от пика оценить для каждой из трех базовых операций, какой процент от пиковой производительности устройства составляет максимальная достигаемая в тесте производительность
4.2 Процент от достижимой производительности
аналогично оценить для каждой операции процент от максимально достижимой производительности с учетом пропускной способности памяти.
Приложение1: исходный текст программы в отдельном c/c++ файле
Требования к программе:

1 Программа должна использовать OpenMP или posix threads для многопоточного распараллеливания
2 Солвер должен корректно работать, т.е. показывать быструю сходимость.

Практическое самостоятельное задание № 2
Распределенная реализация солвера CG для СЛАУ с разреженной матрицей, заданной в формате ELLPACK
Требования к отчету:
Титульный лист, содержащий
1.1 Название курса
1.2 Название задания
1.3 Фамилию, Имя, Отчество(при наличии)
1.4 Номер группы
1.5 Дата подачи
Содержание отчета:
2. Краткое описание задания и программной реализации
2.1 Краткое описание задания
2.2 Краткое описание программной реализации как организованы данные, какие функции реализованы (название, аргументы,назначение)
Просьба указывать, как программа запускается с какими параметрами, с описанием этих параметров.
3. Исследование производительности
3.1 Характеристики вычислительной системы: описание одной или нескольких систем, на которых выполнено исследование.
Использование кластера в этом задании обязательно.
Просьба указывать здесь или в следующих пунктах, как программа компилировалась (каким компилятором, с какими параметрами).
3.2 Результаты измерений производительности
3.2.1Сравнение MPI с OpenMP на многоядерном процессоре.
Для каждой из трех базовых операций и для всего алгоритма солвера сравнить ускорения на разном числе ядер, полученные вMPI и OpenMP режиме, оценить параллельную эффективность. Достаточно одного размера системы, $N=10^6$. Данные представить в виде таблицы.
3.2.2. Параллельное ускорение Измерить MPI ускорение для различных N порядка 10^5, 10^6,,... для каждой из 3-х базовых операций и для всего алгоритма солвера: при фиксированном числе N варьируется число процессов и измеряется параллельное ускорение. Построить графики ускорения.
3.2.3. Масштабирование Измерить масштабирование для различных фиксированных N/P порядка 10^4, 10^5, 10^6. Здесь N/P – количество неизвестных на процесс. В этом тесте число N варьируется пропорционально числу процессов, P.
Данные представить в виде таблицы и графика.

Приложение1: исходный текст программы в отдельном c/c++ файле
Требования к программе:
1 Программа должна использовать MPI для распараллеливания с распределенной памятью, OpenMP или posix threads для многопоточного распараллеливания (которое уже имеется из 1 -го задания)
2 Солвер должен корректно работать, т.е. показывать быструю сходимость.
3 Распараллеливание должно быть корректно.

6.2. Типовые контрольные задания или иные материалы для проведения промежуточной аттестации по дисциплине, критерии и шкалы оценивания

Вопросы к экзамену

1. Параллелизм процессорного ядра и основные характеристики. Конвейер, out-of-order, суперскалярность и другие.
2. Основные характеристики суперкомпьютера - производительность пиковая, теоретически достижимая, пропускные способности.
3. Основные характеристики параллельного алгоритма – оценка сложности, вычислительная интенсивность, ускорение, эффективность, масштабируемость.
4. Устройство оперативной памяти, ключевые характеристики, снижение потерь и оптимизация доступа (DDR, тайминги, иерархия памяти и т. д.)
5. Многопроцессорные системы с общей памятью (UMA, NUMA, ccNUMA), устройство многопроцессорного узла, снижение потерь на NUMA эффекты.
6. Многопоточное распараллеливание Posix, OpenMP, MIMD, общие принципы и проблемы (race condition, false sharing, NUMA, affinity), способы устранения зависимости по данным в сеточных методах.
7. Многопроцессорные системы с распределенной памятью. Устройство и основные характеристики (производительность, латентность и пропускная способность сети, оценка времени передачи данных, топология системы и отображение группы процессов на ресурсы).
8. Распараллеливание с распределенной памятью, MIMD, MPI, основные принципы, виды обмена данными (p2p, групповые обмены различных видов), синхронные и асинхронные обмены
9. Геометрический параллелизм, рациональная декомпозиция расчетной сетки, организация обмена данными. Способы снижения накладных расходов (DMA, overlap, двухуровневая декомпозиция).
10. Устройство графического процессора, общие принципы, архитектура GPU, парадигма потоковой обработки, основные отличия от CPU.
11. Потоковое распараллеливание, OpenCL или CUDA, общие принципы и подходы к реализации (иерархия памяти, буфера, kernel, очереди заданий, накладные расходы)
12. Гетерогенные вычисления. Общие принципы, основные проблемы, способы реализации, многоуровневая декомпозиция.
13. Форматы хранения разреженных матриц (CSR, ELLPACK, ...), особенности операции SpMV и ее распараллеливания, способы повышения производительности (RCM reordering, coalescing, и др.)
14. Параллельное ускорение, параллельная эффективность, масштабирование, закон Амдаля, причины сверхлинейного ускорения
15. Барьер. Схемы реализации и оценки времени выполнения. Реализация барьера на основе синхронных обменов. Реализация барьера на основе семафоров Дейкстры.
16. Методы статической и динамической балансировки загрузки процессоров.
17. Метод сдваивания.
18. Метод коллективного решения.
19. Метод конвейерного параллелизма.
20. Метод диффузной балансировки загрузки.
21. Метод серверного параллелизма.
22. Отладка параллельных приложений, выполнение которых сопровождается недетерминированным потоком сообщений.
23. Параллельные алгоритмы сортировки данных. Слияние методом сдваивания, двустороннее слияние. Сети сортировки. Параллельные алгоритмы сортировки для гибридных вычислительных систем.
24. Параллельные алгоритмы генерации псевдослучайных чисел. Требования к генераторам псевдослучайных чисел. Линейно-конгруэнтные генераторы. М-

последовательности. Достоинства и недостатки генераторов случайных и псевдослучайных чисел.
25. Иерархические алгоритмы разбиения графов. Локальное уточнение.
26. Метод спектральной бисекции графов.
27. Инкрементный алгоритм декомпозиции графов.
28. Адаптивное интегрирование. Метод локального стека. Метод глобального стека.
29. Отказоустойчивые алгоритмы для многопроцессорных вычислительных систем. Локальные контрольные точки. Локальное исправление ошибок при решении гиперболических систем уравнений.

Примеры задач

1. Напишете алгоритм для вычислительной системы с распределенной памятью, определяющий сумму элементов массива. Считайте, что массив уже распределен между mpi-процессами. Оцените ускорение и эффективность алгоритма в предположении, что время суммирования двух элементов равно TauC, а время пересылки одного числа между процессорами равно TauS.
2. Напишете алгоритм для вычислительной системы с общей памятью, определяющий сумму элементов массива. Оцените ускорение и эффективность алгоритма в предположении, что время суммирования двух элементов равно TauC, а минимальное время обращения к семафору равно TauS.
3. Нарисуйте сеть четно-нечетного слияния Бетчера для двух упорядоченных групп элементов, содержащих 4 и 7 элементов. Чему равно число тактов работы построенной сети?
4. Определить период генератора u[i+1]=(5*u[i+1]+3) mod 1024.
5. Определить значение элемента номер 51 последовательности трёхразрядных двоичных чисел, формируемых генератором x^i mod (x^4+ x^3+ 1).
6. Дано: u(t=0;i=0…n)=0, u(t,i=0)=0; u(t=0,…,m;i=n)=1, u(t+1;i=1,...,n-1)=f(u(t,i-1),u(t,i),u(t,i+1)). Напишите параллельную программу для вычислительной системы с распределенной памятью, определяющую значения u(t=m;i=0,...,n). Приведите оценку ускорения и эффективности программы в предположении, что время однократного вычисления функции f равно TauC, а время пересылки одного числа между процессорами равно TauS. Укажите, какой вид параллелизма использован.
7. Напишите алгоритм барьера на основе синхронных обменов.

ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ результатов обучения (РО) по дисциплине				
Оценка / виды оценочных средств	2 (не зачтено)	3 (зачтено)	4 (зачтено)	5 (зачтено)
Знания *(виды оценочных средств: опрос, тесты)*	Отсутствие знаний	Фрагментарные знания	Общие, но не структурированные знания	Сформированные систематические знания
Умения *(виды оценочных средств: практические задания)*	Отсутствие умений	В целом успешное, но не систематическое умение	В целом успешное, но содержащее отдельные пробелы умение (допускает неточности непринципиального характера)	Успешное и систематическое умение
Навыки	Отсутствие	Наличие	В целом,	Сформированные

(владения, опыт деятельности) *(виды оценочных средств: выполнение и защита курсовой работы, отчет по практике, отчет по НИР и т.п.)*	навыков (владений, опыта)	отдельных навыков (наличие фрагментарного опыта)	сформированные навыки (владения), но используемые не в активной форме	навыки (владения), применяемые при решении задач

7. Ресурсное обеспечение:
7.1. Перечень основной и дополнительной литературы

Основная учебно-методическая литература и перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. Якобовский М.В. Введение в параллельные методы решения задач: Учебное пособие / Предисл.: В. А. Садовничий. – М.: Издательство Московского университета, 2012. – 328 с., илл. – (Серия «Суперкомпьютерное образование»), ISBN 978-5-211-06382-2 URL: http://lira.imamod.ru/ITTPMOPS/
2. Лацис А.О. Параллельная обработка данных. – М.: Академия, 2010 г. 336 стр. ISBN 978-5-7695-5951-8
3. Дейкстра Э. Взаимодействие последовательных процессов. http://khpi-iip.mipk.kharkiv.edu/library/extent/dijkstra/ewd123/index.html
4. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. - СПб.: БХВ-Петербург, 2002. URL: http://a1308.ru/books/section/179
5. Якобовский М.В., Кулькова Е.Ю. Решение задач на многопроцессорных вычислительных системах с разделяемой памятью. - М.: СТАНКИН, 2004. – 30 с. http://www.imamod.ru/~serge/arc/stud/Jackob_2.pdf
6. Дональд Э. Кнут Искусство программирования. Том 3. Сортировка и поиск
7. В. Е. Карпов, А. И. Лобанов, Численные методы, алгоритмы и программы. Введение в распараллеливание [Текст] : учебное пособие для студентов высших учебных заведений, обучающихся по направлению подготовки "Прикладные математика и физика" /. - Москва : Физматкнига, 2014. - 190 с. : ил.; 21 см. - (Серия Суперкомпьютерное образование : СКО).; ISBN 978-5-89155-234-0
8. Головизнин В.М., Зайцев М.А., Карабасов С.А., Короткин И.А. Новые алгоритмы вычислительной гидродинамики для многопроцессорных вычислительных комплексов: Монография / Предисл.: В. А. Садовничий. - М.: Издательство Московского университета. 2013. - 472 с., илл.; 8 с. (цв. вклейка). - (Серия «Суперкомпьютерное образование») ISBN 978-5-211-06426-3
9. Гергель В.П., Сысоев А.В. и др., Высокопроизводительные параллельные вычисления. 100 заданий для расширенного лабораторного практикума, Издательство: Физматлит, ISBN: 5922118161, ISBN-13(EAN): 9785922118163, 248 с., 2018
10. Стивенс У. UNIX: взаимодействие процессов: СПб.:Питер; 2002, 576с. Ил.

11. Optimizing software in C++. An optimization guide for Windows, Linux and Mac platforms By Agner Fog. Technical University of Denmark. Copyright © 2004 -2018. Last updated 2018-08-18. https://www.agner.org/optimize/optimizing_cpp.pdf
12. MPI: A Message-Passing Interface StandardVersion 3.0. 2012. https://www.mpi-forum.org/docs/mpi-3.0/mpi30-report.pdf
13. The Argonne Training Program on Extreme-Scale Computing (ATPESC) 2018 https://extremecomputingtraining.anl.gov/archive/
14. Pavan Balaji, et. al. Advanced MPI Programming. Tutorial at SC17? 2017. https://www.mcs.anl.gov/~thakur/sc17-mpi-tutorial/slides.pdf
15. OpenMP tutorial. Blaise Barney, Lawrence Livermore National Laboratory. https://computing.llnl.gov/tutorials/openMP/
16. Спецификации стандарта OpenMP. https://www.openmp.org/specifications/
17. Tim Mattson, Larry Meadows. A “Hands-on” Introduction to OpenMP. https://www.openmp.org/wp-content/uploads/omp-hands-on-SC08.pdf
18. Спецификации стандарта OpenCL https://www.khronos.org/opencl/
19. Демонстрационные примеры на OpenCL NVIDIA, https://developer.nvidia.com/opencl
20. OpenCL Best Practice Guide. NVIDIA https://www.cs.cmu.edu/afs/cs/academic/class/15668-s11/www/cuda-doc/OpenCL_Best_Practices_Guide.pdf
21. AMD Accelerated Parallel Processing OpenCL. Optimization Guide. 2014. https://developer.amd.com/wordpress/media/2013/12/AMD_OpenCL_Programming_Optimization_Guide.pdf
22. Горобец А. В. Параллельная технология численного моделирования задач газовой динамики алгоритмами повышенной точности, Журнал вычислительной математики и математической физики, т. 55, №4, с. 641-652, 2015.

Дополнительная учебно-методическая литература и перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. Лацис А.О. Как построить и использовать суперкомпьютер. - Бестселлер, 2003 г. 274 стр. ISBN 5-98158-003-8
2. Тель Ж. Введение в распределенные алгоритмы. пер. с англ. В. А. Захарова. - М. : МЦНМО, 2009. - 616 с. : ил.
3. Эндрюс Г.Р. Основы многопоточного параллельного и распределенного программирования. — М.: Издательский дом «Вильямс», 2003. — 512 с.: ил. — Парал. тит. англ.
4. Ким А. К., Перекатов В. И., Ермаков С. Г. Микропроцессоры и вычислительные комплексы семейства «Эльбрус». — СПб.: Питер, 2013. - 272 с, ил. — ISBN 978-5-459-01697-0.
5. Михайлов Г.А., Войтишек А.В. Численное статистическое моделирование. Методы Монте-Карло Учебное пособие для студентов вузов. — М.: Академия, 2006. — 368 с. — (Университетский учебник : Прикладная математика и информатика).
6. Зорин А.В., Федоткин М.А., Методы Монте-Карло для параллельных вычислений. Учебное пособие, - М.: Издательство Московского университета, 192 с., 2013 ISBN 978-5-211-06530-7
7. A.Gorobets, S.Soukov, P.Bogdanov. Multilevel parallelization for simulating turbulent flows on most kinds of hybrid supercomputers. Computers and Fluids. Volume 173, Pages 171-177. 2018. https://doi.org/10.1016/j.compfluid.2018.03.011
8. Стронгин, Р.Г. Параллельные вычисления в задачах глобальной оптимизации / Р.Г. Стронгин Р.Г., В.П. Гергель, В.А. Гришагин, К.А. Баркалов. — М.: Изда-тельство Московского университета, 2013. — 280 с.
9. X.Alvarez, A.Gorobets, F.X.Trias, R.Borrell, and G.Oyarzun. HPC2 - a fully portable algebra-dominant framework for heterogeneous computing. Application to CFD. Computers

and Fluids. Volume 173. Pages 285-292. 2018. https://doi.org/10.1016/j.compfluid.2018.01.034
10. Суков С. А., Горобец А. В., Богданов П. Б. Переносимое решение для моделирования сжимаемых течений на всех существующих гибридных суперкомпьютерах, Математическое моделирование, 2017 год, том 29, номер 8, стр. 3-16
11. Горобец А. В. Методика выполнения крупномасштабных расчетов задач газовой динамики, Математическое моделирование (РИНЦ), 2016, том 28, номер 4, стр. 77-91.
12. С. А. Суков, А. В. Горобец, П. Б. Богданов, Адаптация и оптимизация базовых операций газодинамического алгоритма на неструктурированных сетках для расчетов на массивно-параллельных ускорителях, Журнал вычислительной математики и математической физики, 2013, том 53, № 8, с. 1360–1373.

7.2. Перечень лицензионного программного обеспечения, в том числе отечественного производства

При реализации дисциплины может быть использовано следующее программное обеспечение:

1. Операционная система Microsoft Windows 10 Education академическая лицензия

7.3. Перечень профессиональных баз данных и информационных справочных систем

1. http://www.edu.ru – портал Министерства образования и науки РФ
2. http://www.ict.edu.ru – система федеральных образовательных порталов «ИКТ в образовании»
3. http://www.openet.ru - Российский портал открытого образования
4. http://www.mon.gov.ru - Министерство образования и науки Российской Федерации
5. http://www.fasi.gov.ru - Федеральное агентство по науке и инновациям

7.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. Math-Net.Ru [Электронный ресурс] : общероссийский математический портал / Математический институт им. В. А. Стеклова РАН ; Российская академия наук, Отделение математических наук. - М. : [б. и.], 2010. - Загл. с титул. экрана. - Б. ц.
URL: http://www.mathnet.ru

2. Университетская библиотека Online [Электронный ресурс] : электронная библиотечная система / ООО "Директ-Медиа" . - М. : [б. и.], 2001. - Загл. с титул. экрана. - Б. ц. URL: www.biblioclub.ru

3. Универсальные базы данных East View [Электронный ресурс] : информационный ресурс / East View Information Services. - М. : [б. и.], 2012. - Загл. с титул. экрана. - Б. ц.
URL: www.ebiblioteka.ru

4. Научная электронная библиотека eLIBRARY.RU [Электронный ресурс] : информационный портал / ООО "РУНЭБ" ; Санкт-Петербургский государственный университет. - М. : [б. и.], 2005. - Загл. с титул. экрана. - Б. ц. URL: www.eLibrary.ru

7.5. Описание материально-технического обеспечения.

Филиал МГУ в г. Дубне, ответственный за реализацию данной Программы, располагает соответствующей материально-технической базой, включая современную вычислительную технику, объединенную в локальную вычислительную сеть, имеющую выход в Интернет. Используются специализированные компьютерные классы, оснащенные современным оборудованием.

Материальная база подразделения соответствует действующим санитарно-техническим нормам и обеспечивает проведение всех видов занятий (лабораторной, практической, дисциплинарной и междисциплинарной подготовки) и научно-исследовательской работы обучающихся, предусмотренных учебным планом.

8. Методические рекомендации по организации изучения дисциплины

8.1. Формы и методы преподавания дисциплины

(перечисляются в соответствии с таблицей 5.1.)

Используемые формы и методы обучения:

лекции

семинарские занятия

самостоятельная работа студентов.

В процессе преподавания дисциплины преподаватель использует как классические формы и методы обучения (лекции и семинарские занятия), так и активные методы обучения.

При проведении лекционных занятий преподаватель использует аудиовизуальные, компьютерные и мультимедийные средства обучения, а также демонстрационные и наглядно-иллюстрационные (в том числе раздаточные) материалы.

Семинарские занятия проводятся в форме проблемной ситуации, когда некоторый аспект рассмотренной темы излагается преподавателем более подробно. Часть информации конспектируется. Большая часть времени выделена на работу с использованием компьютерной техники и программного обеспечения.

В рамках курса используются активные и интерактивные методы обучения в процессе проведения занятий. Основными особенностями интерактивных занятий являются интерактивные практические упражнения и задания, которые выполняются обучающимися не только и не столько на закрепление изученного материала, но и на самостоятельное изучение нового.

8.2. Методические рекомендации преподавателю

Перед началом изучения дисциплины преподаватель должен ознакомить студентов с видами учебной и самостоятельной работы, перечнем литературы и интернет-ресурсов, формами текущей и промежуточной аттестации, с критериями оценки качества знаний для итоговой оценки по дисциплине.

При проведении лекций, преподаватель:

1) формулирует тему и цель занятия;

2) излагает основные теоретические положения;

3) с помощью мультимедийного оборудования и/или под запись дает определения основных понятий, расчетных формул;

4) проводит примеры из отечественного и зарубежного опыта, дает текущие статистические данные для наглядного и образного представления изучаемого материала;

5) в конце занятия дает вопросы для самостоятельного изучения.

Для семинарских занятий

Подготовка к проведению занятий проводится регулярно. Организация преподавателем семинарских занятий должна удовлетворять следующим требования: количество занятий должно соответствовать учебному плану программы, содержание планов должно соответствовать программе, план занятий должен содержать перечень рассматриваемых вопросов.

Во время семинарских занятий используются словесные методы обучения, как беседа и дискуссия, что позволяет вовлекать в учебный процесс всех слушателей и стимулирует творческий потенциал обучающихся.

При подготовке семинарскому занятию преподавателю необходимо знать план его проведения, продумать формулировки и содержание учебных вопросов, выносимых на обсуждение.

В начале занятия преподаватель должен раскрыть теоретическую и практическую значимость темы занятия, определить порядок его проведения, время на обсуждение каждого учебного вопроса. В ходе занятия следует дать возможность выступить всем желающим и предложить выступить тем слушателям, которые проявляют пассивность.

Целесообразно, в ходе обсуждения учебных вопросов, задавать выступающим и аудитории дополнительные и уточняющие вопросы с целью выяснения их позиций по существу обсуждаемых проблем, а также поощрять выступление с места в виде кратких дополнений. На занятиях проводится отработка практических умений под контролем преподавателя

8.3. Методические рекомендации студентам по организации самостоятельной работы.

Приступая к изучению новой учебной дисциплины, студенты должны ознакомиться с учебной программой, учебной, научной и методической литературой, имеющейся в библиотеке университета, встретиться с преподавателем, ведущим дисциплину, получить в библиотеке рекомендованные учебники и учебно-методические пособия, осуществить запись на соответствующий курс в среде электронного обучения университета.

Глубина усвоения дисциплины зависит от активной и систематической работы студента на лекциях и практических занятиях, а также в ходе самостоятельной работы, по изучению рекомендованной литературы.

На лекциях важно сосредоточить внимание на ее содержании. Это поможет лучше воспринимать учебный материал и уяснить взаимосвязь проблем по всей дисциплине. Основное содержание лекции целесообразнее записывать в тетради в виде ключевых фраз, понятий, тезисов, обобщений, схем, опорных выводов. Необходимо обращать внимание на термины, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации. Желательно оставлять в конспектах поля, на которых делать пометки из рекомендованной литературы, дополняющей материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений. С целью уяснения теоретических положений, разрешения спорных ситуаций необходимо задавать преподавателю уточняющие вопросы. Для закрепления содержания лекции в памяти, необходимо во время самостоятельной работы внимательно прочесть свой конспект и дополнить его записями из учебников и рекомендованной литературы. Конспектирование читаемых лекций и их последующая доработка способствует более глубокому усвоению знаний, и поэтому являются важной формой учебной деятельности студентов.

Методические указания для самостоятельной работы обучающихся

Прочное усвоение и долговременное закрепление учебного материала невозможно без продуманной самостоятельной работы. Такая работа требует от студента значительных усилий, творчества и высокой организованности. В ходе самостоятельной работы студенты выполняют следующие задачи: дорабатывают лекции, изучают рекомендованную литературу, готовятся к практическим занятиям, к коллоквиуму, контрольным работам по отдельным темам дисциплины. При этом эффективность учебной деятельности студента во многом зависит от того, как он распорядился выделенным для самостоятельной работы бюджетом времени.

Результатом самостоятельной работы является прочное усвоение материалов по предмету согласно программы дисциплины. В итоге этой работы формируются профессиональные умения и компетенции, развивается творческий подход к решению возникших в ходе учебной деятельности проблемных задач, появляется самостоятельности мышления.

Решение задач

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения

проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи).

Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками.

Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом.

Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты.

Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

Задача — это цель, заданная в определенных условиях, решение задачи — процесс достижения поставленной цели, поиск необходимых для этого средств.

Алгоритм решения задач:

1. Внимательно прочитайте условие задания и уясните основной вопрос, представьте процессы и явления, описанные в условии.

2. Повторно прочтите условие для того, чтобы чётко представить основной вопрос, проблему, цель решения, заданные величины, опираясь на которые можно вести поиски решения.

3. Произведите краткую запись условия задания.

4. Если необходимо составьте таблицу, схему, рисунок или чертёж.

5. Определите метод решения задания, составьте план решения.

6. Запишите основные понятия, формулы, описывающие процессы, предложенные заданной системой.

7. Найдите решение в общем виде, выразив искомые величины через заданные.

9. Проверьте правильность решения задания.

10. Произведите оценку реальности полученного решения.

11. Запишите ответ.

9. Разработчик (разработчики) программы.

к.т.н. Григорьева М.А.