

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

С1.2.12.2 СУПЕРКОМПЬЮТЕРНЫЕ ВЫЧИСЛЕНИЯ

Специальность **01.05.01** **Фундаментальные математика и механика**

Специализация **Вычислительная математика и вычислительная механика**

Квалификация (степень) выпускника – **Математик. Механик. Преподаватель**

Форма обучения **очная**

Пенза, 2016

1. Цели освоения дисциплины «Суперкомпьютерные вычисления»

Целями освоения учебной дисциплины «Суперкомпьютерные вычисления» являются обучение студентов работе на суперкомпьютерных вычислительных системах, привитие студентам навыков решения больших физико-технических и инженерно-физических задач, заложить понимание формальных основ теории параллельных вычислений и повысить у студентов уровень научной интуиции, необходимой для формализации содержательных задач прикладных наук и адаптации алгоритмов их решения для суперкомпьютерных вычислительных систем.

2. Место дисциплины в структуре ОПОП специалитета

Дисциплина «Суперкомпьютерные вычисления» в учебном плане находится в вариативной части цикла С1 и является одной из дисциплин, формирующих профессиональные знания и навыки, характерные для специалиста по специальности 01.05.01 «Фундаментальная математика и механика» (Специализация «Вычислительная математика и вычислительная механика»).

Изучение данной дисциплины базируется на знаниях на знаниях курса «Технология программирования и работа на ЭВМ»

Основные положения дисциплины должны быть использованы в дальнейшем при изучении следующих дисциплин:

- учебная практика 1

3. Компетенции обучающегося, формируемые в результате освоения дисциплины «Суперкомпьютерные вычисления»

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО по данной специальности:

Коды компетенции	Наименование компетенции	Структурные элементы компетенции (в результате освоения дисциплины обучающийся должен знать, уметь, владеть)
1	2	3
ОПК-4	способность находить, анализировать, реализовывать программно и использовать на практике математические алгоритмы, в том числе с применением современных вычислительных систем	Знать: математические алгоритмы, в том числе с применением современных вычислительных систем
		Уметь: анализировать и реализовывать математические алгоритмы
		Владеть: методами и технологиями разработки программ для задач из указанных разделов
ПК-1	способность к самостоятельному анализу поставленной задачи, выбора корректного метода ее решения, построения алгоритма и его реализация, обработке и анализу полученной информации	Знать: способы построения алгоритмов и их реализации
		Уметь: анализировать поставленную задачу
		Владеть: методами построения и реализации алгоритмов
ПК-5	способность использовать методы математического и алгоритмического	Знать: методы математического и алгоритмического моделирования при решении теоретических и прикладных задач

	моделирования при решении теоретических и прикладных задач	Уметь: анализировать и реализовывать методы математического и алгоритмического моделирования
		Владеть: методами и технологиями разработки программ для задач из указанных разделов
СК-1	способность разрабатывать параллельные алгоритмы для решения задач вычислительной математики и вычислительной механики	Знать: параллельные алгоритмы для решения задач вычислительной математики и вычислительной механики
		Уметь: реализовывать параллельные алгоритмы
		Владеть: методами и технологиями разработки программ для задач вычислительной математики и вычислительной механики
СК-2	способность разрабатывать и реализовывать на суперкомпьютерах и распределенных вычислительных системах численные методы решения задач математики и естествознания	Знать: правила компиляции, отладки и запуска готовых программ на вычислительных кластерах; иметь представление о существующих методах позволяющих работать и создавать параллельные программы
		Уметь: реализовывать методы решения задач математики и естествознания
		Владеть: методами и технологиями разработки программ для суперкомпьютерах и распределенных вычислительных систем

4. Структура и содержание дисциплины «Суперкомпьютерные вычисления»

4.1. Структура дисциплины

Общая трудоемкость дисциплины составляет 5 зачетных единиц, 180 часов.

Количество аудиторных занятий: лекции – 51 час, лабораторные занятия – 85 часов, самостоятельная работа: 44 часа.

№ п/п	Наименование разделов и тем дисциплины	Семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)						Формы текущего контроля успеваемости (по неделям семестра)	
				Аудиторная работа			Самостоятельная работа			Коллоквиум	Проверка лаб. работ
				Всего	Лекции	Лабораторные занятия	Всего	Подготовка к ауд. занятиям	Подготовка к экзамену		
1.	Раздел 1. Архитектура суперкомпьютерных вычислительных систем.	8	1-2	8	4	4		0.5			
2.	Раздел 2. Операционная система Linux. Процессы и потоки.	8	3-6								
	Тема 2.1. Процессы и потоки.	8	3-4	8	4	4		0.5			
	Тема 2.2. Параллелизм на уровне данных.	8	5-6	8	4	4		0.5		6	
3.	Раздел 3. Технология MPI. Простейшая MPI-программа. Пересылка данных между двумя процессами.	8	7-17								
	Тема 3.1. Организация параллельной части программы.	8	7-8	8	4	4		0.5			
	Тема 3.2. Пересылка данных «точка - точка»	8	9-10	8	4	4		0.5			
	Тема 3.3. Блокировка процессов.	8	11-12	8	4	4		0.5		12	
	Тема 3.4. Скалярное произведение векторов и организация пересылки данных	8	13-17	20	10	10		1		17	

4.	Раздел 4. Технология MPI. Коллективное взаимодействие процессов.	9	1-8							8
	Тема 4.1. Коллективное взаимодействие и блокировка процессов.	9	1-4	16	4	12		1		
	Тема 4.2. Выделение памяти.	9	5-8	16	4	12		1		
5.	Раздел 5. Технология MPI. Работа с большими объемами данных	9	9-13	20	5	15		1		13
6.	Раздел 6. Повышение производительности программ.	9	14-17	16	4	12		1		17
	<i>Подготовка к экзамену</i>	9							36	
	Общая трудоемкость, в часах			136	51	85	44	8	36	
							Промежуточная аттестация			
							Форма		Семестр	
							Зачет		8	
							Экзамен		9	

4.2. Содержание дисциплины

1. Архитектура суперкомпьютерных вычислительных систем.
2. Многопроцессорные вычислительные системы с распределенной памятью.
3. Введение в операционную систему Linux.
4. Компиляция программ, отладчик.
5. Процессы и потоки в системе Linux.
6. Запуск параллельной программы.
7. Организация параллельной обработки на уровне данных.
8. Введение в технологию MPI.
9. Простейшая MPI-программа.
10. Получение номера процесса и количество процессов.
11. Коммуникаторы и группы.
12. Пересылка данных с блокировкой и без блокировки процессов.
13. Коллективное взаимодействие процессов.
14. Выделение динамической памяти.
15. Методы повышения скорости работы с динамической памятью.
16. Анализ производительности параллельной программы.
17. Внутренний параллелизм на примере решения системы линейных уравнений методом Гаусса.
18. Внутренний параллелизм на примере решения системы линейных уравнений методом простой итерации.
19. Внутренний параллелизм на примере решения системы линейных уравнений с трехдиагональной матрицей.
20. Способы повышения производительности умножения матриц.
21. Внутренний параллелизм вычисления произведения $A^T(B - B^T)A$.
22. Оценка эффективности параллельной части программы. Факторы снижающие производительность вычислительных комплексов.

5. Образовательные технологии

Активные и интерактивные формы: лекции, лабораторные занятия, зачет и экзамен. В течение семестров студенты выполняют лабораторные работы, указанные преподавателем.

6. Учебно-методическое обеспечение самостоятельной работы студентов. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

Выполнение лабораторных работ оцениваются по рейтинговой системе. Экзамены оцениваются по системе: неудовлетворительно, удовлетворительно, хорошо, отлично.

6.1. План самостоятельной работы студентов

№ нед.	Тема	Вид самостоятельной работы	Задание	Рекомендуемая литература	Количество часов
1-2, 8сем .	Архитектура суперкомпьютерных вычислительных систем	<i>Подготовка к защите лабораторной работы</i>	Изучить архитектуру суперкомпьютерных вычислительных систем	Гергель, В.П. Теория и практика параллельных вычислений [Электронный ресурс] : учеб. пособие — Электрон. дан. — Москва :	0.5

				, 2016. — 500 с.	
3-6, 8сем .	Операционная система Linux. Процессы и потоки.	<i>Подготовка к защите лабораторной работы</i>	Изучить структуру MPI-программы: аргументы функций main, MPI_Init, MPI_Finalize, MPI_Comm_rank, MPI_Comm_size. Произвести компиляцию и запуск программы, опции команд mpicc, mpicxx, mpirun.	Гергель, В.П. Теория и практика параллельных вычислений [Электронный ресурс] : учеб. пособие — Электрон. дан. — Москва : , 2016. — 500	1
7-17, 8сем .	Технология MPI. Простейшая MPI-программа. Передача данных между двумя процессами.	<i>Подготовка к защите лабораторной работы</i>	Написать MPI-программу нахождения скалярного произведения векторов, используя функции точечного обмена MPI.	Гергель, В.П. Теория и практика параллельных вычислений [Электронный ресурс] : учеб. пособие — Электрон. дан. — Москва : , 2016. — 500	2.5
11-18, 9сем .	Технология MPI. Коллективное взаимодействие процессов.	<i>Подготовка к защите лабораторной работы</i>	Используя технологию MPI реализовать метод простой итерации решения системы линейных алгебраических уравнений $Ax=b$. Построить график ускорения времени работы программы от количества задействованных MPI процессов.	Гергель, В.П. Теория и практика параллельных вычислений [Электронный ресурс] : учеб. пособие — Электрон. дан. — Москва : , 2016. — 500	2
1-8, 9сем .	Технология MPI. Работа с большими объемами данных	<i>Подготовка к защите лабораторной работы</i>	Исследовать внутренний параллелизм явной схемы решения уравнения теплопроводности. Написать параллельную программу реализующую явную схему.	Гергель, В.П. Теория и практика параллельных вычислений [Электронный ресурс] : учеб. пособие — Электрон. дан. — Москва : , 2016. — 500	1
9-13, 9сем .	Повышение производительности программ.	<i>Подготовка к защите лабораторной работы</i>	Написать программу, реализующую умножения двух квадратных матриц. Исследо-	Гергель, В.П. Теория и практика параллельных вычислений [Электрон-	1

			вать внутренний параллелизм умножения матриц. Проверить, как влияет на время работы, изменения порядка выполнения циклов по i,j,k. Для компиляции программы использовать параметры -O2, -O3, -O4.	ный ресурс] : учеб. пособие — Электрон. дан. — Москва : , 2016. — 500	
9 сем		Подготовка к экзамену			36

6.2. Методические указания по организации самостоятельной работы студентов

Студенты получают от преподавателя задание на повторение пройденного материала и самостоятельное изучение дополнительного материала по изучаемым темам лекционного курса. Преподаватель предлагает студентам литературу для самостоятельного изучения, а также выдает дополнительные практические задания (списки задач из учебников и сборников задач согласно списку основной и дополнительной литературы по изучаемой дисциплине).

6.3. Материалы для проведения текущего и промежуточного контроля знаний студентов

Контроль освоения компетенций

№ п/п	Вид контроля	Контролируемые темы (разделы)	Компетенции, компоненты которых контролируются
1	лабораторная работа, зачет. экзамен	Архитектура суперкомпьютерных вычислительных систем	ОПК4, ПК1,5, СК1,2
2	лабораторная работа, зачет. экзамен	Операционная система Linux. Процессы и потоки.	ОПК4, ПК1,5, СК1,2
3	лабораторная работа, зачет. экзамен	Технология MPI. Простейшая MPI-программа. Пересылка данных между двумя процессами.	ОПК4, ПК1,5, СК1,2
4	лабораторная работа, зачет. экзамен	Технология MPI. Коллективное взаимодействие процессов.	ОПК4, ПК1,5, СК1,2
5	лабораторная работа, зачет. экзамен	Технология MPI. Работа с большими объемами данных	ОПК4, ПК1,5, СК1,2
6	лабораторная работа, зачет. экзамен	Повышение производительности программ.	ОПК4, ПК1,5, СК1,2

Лабораторная работа № 1

Тема работы: «MPI-программа Hello World. Процессы в OS Linux»

Задание. Изучить структуру MPI-программы: аргументы функций main, MPI_Init, MPI_Finalize, MPI_Comm_rank, MPI_Comm_size. Произвести компиляцию и запуск программы, опции команд mpicc, mpicxx, mpirun.

Лабораторная работа № 2

Тема работы: «MPI-функции точечного обмена»

Задание. Написать MPI-программу нахождения скалярного произведения векторов, используя функции точечного обмена MPI.

Лабораторная работа № 3

Тема работы: «MPI-функции коллективного взаимодействия процессов»

Задание. Написать MPI-программу нахождения скалярного произведения векторов, используя функции коллективного обмена MPI.

Лабораторная работа № 4

Тема работы: «MPI-реализация метода простой итерации»

Задание. Используя технологию MPI реализовать метод простой итерации решения системы линейных алгебраических уравнений $Ax=b$. Построить график ускорения времени работы программы от количества задействованных MPI процессов.

Примечание. Для выделения памяти использовать команду:

```
B = (double*)malloc(m * m * sizeof(double));
```

где m - размер матрицы.

Использовать функцию умножения матрицы на столбец:

```
void matvec (double* B, double* g, double* xold, double* x, int k, int m){  
for (int i = 0; i < k; i ++){  
double a = 0.;  
double* row = B + i * m;  
for(int j = 0; j < m; j ++)  
a += row[j] * xold[j];  
x[i] = -a + g[i];  
}  
}
```

Лабораторная работа № 5

Тема работы: «Повышение производительности операции умножения матриц»

Задание. Написать программу, реализующую умножения двух квадратных матриц. Исследовать внутренний параллелизм умножения матриц. Проверить, как влияет на время работы, изменения порядка выполнения циклов по i, j, k . Для компиляции программы использовать параметры $-O2$, $-O3$, $-O4$.

Лабораторная работа № 6

Тема работы: «Параллельная реализация численного решения уравнения теплопроводности»

Задание. Исследовать внутренний параллелизм явной схемы решения уравнения теплопроводности. Написать параллельную программу реализующую явную схему.

Примечание. Для написания программы можно использовать язык C++, конструктор `std::thread`.

Примерный перечень вопросов к экзамену

1. Принципы построения параллельных вычислительных систем
2. Пути достижения параллелизма
3. Функциональные вычислительные устройства. Многоуровневая и модульная
4. память. Конвейерные и векторные вычисления. Процессорные матрицы.
5. Многопроцессорные вычислительные системы с общей и распределенной памятью
6. (мультимикропроцессоры и мультимикрокомпьютеры). Микропроцессорные системы.
7. Способы построения многопроцессорных вычислительных систем
8. Схемы коммутации (полная коммутация - общая память, перекрестные коммутаторы, локальные схемы коммутации - общая шина, решетки, кластеры).
9. Анализ параллельных алгоритмов и типовые топологии схем коммутации – кольцо,
10. линейка, решетки, полный граф, гиперкуб, тор, дерево. Аппаратная реализация и
11. программная эмуляция топологий.
12. СуперЭВМ. Многопроцессорные вычислительные комплексы (МВС).
13. Многомашинные вычислительные комплексы. Сети ЭВМ.
14. Примеры современных высокопроизводительных вычислительных систем (Cray
15. T932, IBM SP2, HP Exemplar, ASCI RED).
16. Суперкомпьютерные вычислительные системы в России.
17. Концепция неограниченного параллелизма. Граф сдваивания.
18. Модель выполнения MPI программы.
19. Процессы и потоки в ОС Linux. Технологии параллельного программирования.
20. Многопоточное программирование. `std::thread`
21. Способы повышения производительности умножения матриц.
22. Преимущества и недостатки вычислительных систем с общей памятью на примере вычисления элементов матрицы.
23. Технологии программирования вычислительных систем с распределенной памятью.
24. Параллелизм на уровне данных.
25. Получение количества процессоров, доступных для выполнения.
26. Получение номера процесса в группе.
27. Распараллеливание циклов. Стратегия распределения циклов по процессам.
28. Организация передачи сообщений. Скалярное произведение векторов.
29. Компиляция и запуск программ на вычислительном комплексе (Компиляторы, опции компиляторов, make- файл, очередь выполнения и тд...)
30. Внутренний параллелизм на примере решения системы линейных уравнений методом Гаусса.
31. Внутренний параллелизм на примере решения системы линейных уравнений методом простой итерации.
32. Внутренний параллелизм на примере решения системы линейных уравнений с трехдиагональной матрицей.
33. Способы повышения производительности умножения матриц.
34. Внутренний параллелизм вычисления произведения $A^T(B - B^T)A$.
35. Оценка эффективности параллельной части программы. Факторы снижающие производительность вычислительных комплексов.

7. Учебно-методическое и информационное обеспечение дисциплины «Суперкомпьютерные вычисления»

а) основная литература:

1. Гергель, В.П. Теория и практика параллельных вычислений [Электронный ресурс] : учеб. пособие — Электрон. дан. — Москва : , 2016. — 500 с. — Режим доступа: <https://e.lanbook.com/book/100527>. — Загл. с экрана.
2. Гергель, В.П. Технологии построения и использования кластерных систем [Электронный ресурс] : учеб. пособие — Электрон. дан. — Москва : , 2016. — 548 с. — Режим доступа: <https://e.lanbook.com/book/100517>. — Загл. с экрана.

б) дополнительная литература:

1. Воеводин, В.В. Вычислительная математика и структура алгоритмов [Электронный ресурс] : учеб. пособие — Электрон. дан. — Москва : , 2016. — 145 с. — Режим доступа: <https://e.lanbook.com/book/100738>. — Загл. с экрана.

в) Интернет-ресурсы:

1. PARALLEL.RU - Информационно-аналитический центр по параллельным вычислениям.
2. unihub - Технологическая платформа программы «Университетский кластер»

Программное обеспечение: Операционная система Linux.

8. Материально-техническое обеспечение дисциплины «Суперкомпьютерные вычисления»


При освоении дисциплины и выполнения лабораторных работ необходимы учебные аудитории оснащенные персональными компьютерами, локальная сеть, кластер (суперкомпьютер).

9. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

Обучающиеся, из числа инвалидов и лиц с ограниченными возможностями здоровья обеспечены электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья. Форма проведения текущей и промежуточной аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей (устно, письменно на компьютере, в формате тестирования и т.д.) и позволяют оценить достижения ими запланированных в основной образовательной программе результатов обучения и уровня сформированности всех заявленных компетенций. На экзамен приглашается сопровождающий, который обеспечивает техническое сопровождение студенту. При необходимости студенту-инвалиду предоставляется дополнительное время для подготовки ответа на зачете и экзамене.

Рабочая программа дисциплины С1.2.12.2 «Суперкомпьютерные вычисления» составлена в соответствии с требованиями ФГОС ВО по специальности 01.05.01 «Фундаментальные математика и механика».

Программу составили:

1. _____ Долгарев И.А., доцент каф. МСМ 
(Ф.И.О., должность, подпись)

2. _____
(Ф.И.О., должность, подпись)


Настоящая программа не может быть воспроизведена ни в какой форме без предварительного письменного разрешения кафедры-разработчика программы.

Программа одобрена на заседании кафедры МСМ

Протокол № 3

от «30» сентября 2016 года

Зав. кафедрой _____

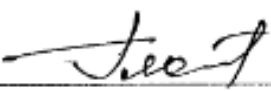

Смирнов Ю.Г.
(подпись, Ф.И.О.)

Программа одобрена методической комиссией факультета ВТ

Протокол № 2

от «3» 10 2016 года

Председатель методической комиссии
факультета ВТ


Глотова Т.В.
(подпись) (Ф.И.О.)

