

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кемеровский государственный университет»

Математический факультет



Рабочая программа дисциплины (модуля)

Б1.В.ДВ.15.2 Параллельные вычислительные алгоритмы

(Наименование дисциплины (модуля))

Направление подготовки

01.03.02 Прикладная математика и информатика

Направленность (профиль) подготовки

Исследование операций и системный анализ

Уровень бакалавриата

Форма обучения

Очная

Кемерово 2015

Рабочая программа дисциплины утверждена Ученым советом факультета
(протокол Ученого совета факультета № 9 от 13. 04. 2015 г.)

Утверждена с обновлениями в части реорганизации структуры факультета
(протокол Ученого совета факультета № 12 от 22. 06. 2015 г.)

Рабочая программа дисциплины одобрена на заседании кафедры ЮНЕСКО по ИВТ

СОДЕЖАНИЕ

1.	Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы	4
2.	Место дисциплины в структуре программы бакалавриата.....	4
3.	Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам занятий) и на самостоятельную работу обучающихся.....	5
3.1.	Объём дисциплины (модуля) по видам учебных занятий (в часах).....	5
4.	Содержание дисциплины (модуля), структурированное по разделам (темам) с указанием отведенного на них количества академических часов и видов учебных занятий	5
4.1.	Разделы дисциплины (модуля) и трудоемкость по видам учебных занятий (в академических часах).....	6
4.2.	Содержание дисциплины (модуля), структурированное по темам (разделам)	6
5.	Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю).....	8
6.	Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю).....	9
6.1.	Паспорт фонда оценочных средств по дисциплине (модулю)	9
6.2.	Типовые контрольные задания или иные материалы	10
7.	Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля).....	17
8.	Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее – сеть «Интернет»), необходимых для освоения дисциплины (модуля)	18
9.	Методические указания для обучающихся по освоению дисциплины (модуля)	18
10.	Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю).....	19
11.	Иные сведения и материалы.....	20
11.1.	Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья.....	20

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

В результате освоения программы бакалавриата обучающийся должен овладеть следующими результатами обучения по дисциплине (модулю):

Коды компетенций	Результаты освоения ООП Содержание компетенций	Перечень планируемых результатов обучения по дисциплине
ОПК-1	способность использовать базовые знания естественных наук, математики и информатики, основные факты, концепции, принципы теорий, связанных с прикладной математикой и информатикой	знать общенаучные базовые знания естественных наук, математики и информатики; уметь демонстрировать эти знания; владеть пониманием основных фактов, концепций, принципов теорий, связанных с прикладной математикой и информатикой;
ПК-2	способность понимать, совершенствовать и применять современный математический аппарат	знать современный математический аппарат; уметь понимать, совершенствовать и применять современный математический аппарат
ПК-11	способность к организации педагогической деятельности в конкретной предметной области (математика и информатика)	знать учебные дисциплины; уметь преподавать
ПК-12	способность к планированию и осуществлению педагогической деятельности с учетом специфики предметной области в общеобразовательных и профессиональных образовательных организациях	уметь применять на практике современные методы педагогики и средства обучения; владеть методикой педагогики

2. Место дисциплины в структуре программы бакалавриата

Дисциплина (модуль) «Б1.В.ДВ.15.2 Параллельные вычислительные алгоритмы» относится к дисциплинам базового цикла и является одной из дисциплин, в рамках которой изучаются технологии и средства параллельного программирования, способы распараллеливания вычислительных алгоритмов.

Данный курс, наряду с курсами по программированию и курсом “Технологии параллельного программирования”, составляет основу образования студента в части параллельных вычислений и направлен на совершенствование навыков разработки параллельных алгоритмов для решения прикладных задач. Курс рассчитан на студентов, имеющих подготовку по предшествующим курсам, касающихся основам программирования с использованием алгоритмического языка С, вычислительным методам, а также технологиям параллельного программирования. В течение преподавания курса предполагается, что студенты знакомы с основными понятиями алгебры, комбинаторики, логики, информатики, которые читаются на факультете перед изучением

данной дисциплины.

Знания, навыки и умения, приобретенные в результате прохождения курса, будут востребованы при выполнении итоговой квалификационной работы, связанной с реализацией высокоеффективных параллельных алгоритмов, математическим моделированием и обработкой результатов экспериментов, решением конкретных задач естественнонаучного направления.

Дисциплина изучается на 3 курсе в 5 семестре.

3. Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам занятий) и на самостоятельную работу обучающихся

Общая трудоемкость (объем) дисциплины (модуля) составляет 3 зачетных единиц (з.е.), 108 академических часов.

3.1. Объем дисциплины (модуля) по видам учебных занятий (в часах)

Объем дисциплины	Всего часов
	для очной формы обучения
Общая трудоемкость дисциплины	108
Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)	
Аудиторная работа (всего):	72
в том числе:	
лекции	18
семинары, практические занятия	
практикумы	54
лабораторные работы	
в т.ч. в активной и интерактивной формах	
Внеаудиторная работа (всего):	36
В том числе, индивидуальная работа обучающихся с преподавателем:	
курсовое проектирование	
групповая, индивидуальная консультация и иные виды учебной деятельности, предусматривающие групповую или индивидуальную работу обучающихся с преподавателем)	
творческая работа (эссе)	
Самостоятельная работа обучающихся (всего)	
Вид промежуточной аттестации обучающегося (зачет / экзамен)	зачет

4. Содержание дисциплины (модуля), структурированное по разделам (темам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины (модуля) и трудоемкость по видам учебных занятий (в академических часах)

№ п/п	Раздел дисциплины	Общая трудоемкость (часах)	Виды учебных занятий, включая самостоятельную работу обучающихся и трудоемкость (в часах)			Формы текущего контроля успеваемости
			аудиторные учебные занятия	самостояте- льная работа обучающих- ся		
		всего	лекции	семинары, практиче- ские занятия		
1.	Анализ параллельных алгоритмов. Принципы разработки параллельных методов.	7	2	3	2	Контрольная работа
2.	Распараллеливание матрично-векторных операций.	35	6	17	12	Отчет по лабораторной работе
3.	Решение систем линейных алгебраических уравнений	32	5	17	10	Отчет по лабораторной работе
4.	Параллельные сортировки и генетические алгоритмы	34	5	17	12	Отчет по лабораторной работе
Всего:		108	18	54	36	

4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Содержание лекционного курса

	Наименование раздела дисциплины	Содержание раздела дисциплины
1	Анализ параллельных алгоритмов. Принципы разработки параллельных методов.	
1.1.	Требования и сложность алгоритма. Критерии качества программного обеспечения	Различные определения алгоритма, требованиям, предъявляемым к алгоритмам. Критерии качества программного обеспечения. Основы теории сложности алгоритмов и вычислений.
1.2.	Модель параллельного алгоритма. Показатели эффективности параллельного алгоритма.	Модель параллельного алгоритма в виде графа "операнд-операция" и расписания, различные определения времени выполнения параллельного алгоритма, основные показатели качества параллельных алгоритмов: ускорение и эффективность. Методика теоретической оценки ускорения и эффективности распараллеливания программ. Формальная модель ускорения. Введенные понятия демонстрируются на ряде учебных примеров.
1.3.	Модели оценки времени выполнения коммуникационных операций	Различные модели оценки времени выполнения коммуникационных операций. Вычисляются константы моделей.
2	Распараллеливание матрично-векторных операций.	

2.1.	Распараллеливание умножения матрицы на вектор	Различные подходы к распределению элементов матрицы и вектора между процессорами вычислительной системы согласно выбранным базовым операциям при распараллеливании. Реализации алгоритмов для вычислительных систем с общей или распределенной памятью, а также для гибридных систем. Вычисление ускорения и эффективности распараллеливания каждого алгоритмов.
2.2.	Распараллеливание умножения матриц	Различные подходы к распределению элементов матриц между процессорами вычислительной системы согласно выбранным базовым операциям при распараллеливании. Реализации алгоритмов для вычислительных систем с общей или распределенной памятью, а также для гибридных систем. Вычисление ускорения и эффективности распараллеливания каждого алгоритмов.
2.3.	Параллельные реализации метода Гаусса	Различные метода Гаусса решения систем линейных алгебраических уравнений для систем с общей и распределенной памятью. Оценка ускорения и эффективности реализаций параллельных алгоритмов.
3	Решение систем линейных алгебраических уравнений.	
3.1	Параллельные реализации метода Гаусса	Различные метода Гаусса решения систем линейных алгебраических уравнений для систем с общей и распределенной памятью. Оценка ускорения и эффективности реализаций параллельных алгоритмов.
4	Параллельные сортировки и генетические алгоритмы	
4.1	Параллельные сортировки	Параллельные алгоритмы сортировки на базе последовательных алгоритмов: сортировки пузырьком и сортировка Шелла. Реализации параллельных алгоритмов для различных технологий распараллеливания. Оценки эффективности распараллеливания построенных алгоритмов.
4.2	Генетические алгоритмы	Основные понятия и методы, лежащие в основе генетических алгоритмов поиска глобального оптимума. Этапы генетического алгоритма поиска оптимума, подходы к распараллеливанию алгоритма, распараллеливание алгоритма для систем с общей и распределенной памятью. Оценки эффективности распараллеливания построенных алгоритмов.

Содержание практикумов

	Наименование раздела дисциплины	Содержание раздела дисциплины
1	Анализ параллельных алгоритмов. Принципы разработки параллельных методов.	
1.3	Модели оценки времени выполнения коммуникационных операций	Различные модели оценки времени выполнения коммуникационных операций. Вычисление констант моделей .
2	Распараллеливание матрично-векторных операций.	
2.1	Распараллеливание умножения матрицы на вектор	Различные подходы к распределению элементов матрицы и вектора между процессорами вычислительной системы согласно выбранным базовым операциям при распараллеливании. Реализации алгоритмов для вычислительных систем с общей или распределенной

		памятью, а также для гибридных систем. Вычисление ускорения и эффективности распараллеливания каждого алгоритмов.
2.2	Распараллеливание умножения матриц	Различные подходы к распределению элементов матриц между процессорами вычислительной системы согласно выбранным базовым операциям при распараллеливании. Реализации алгоритмов для вычислительных систем с общей или распределенной памятью, а также для гибридных систем. Вычисление ускорения и эффективности распараллеливания каждого алгоритмов.
2.3	Параллельные реализации метода Гаусса	Различные методы Гаусса решения систем линейных алгебраических уравнений для систем с общей и распределенной памятью. Оценка ускорения и эффективности реализаций параллельных алгоритмов.
3	Решение систем линейных алгебраических уравнений.	
3.1	Параллельные реализации метода Гаусса	Различные методы Гаусса решения систем линейных алгебраических уравнений для систем с общей и распределенной памятью. Оценка ускорения и эффективности реализаций параллельных алгоритмов.
4	Параллельные сортировки и генетические алгоритмы	
4.1	Параллельные сортировки	Параллельные алгоритмы сортировки на базе последовательных алгоритмов: сортировки пузырьком и сортировка Шелла. Реализации параллельных алгоритмов для различных технологий распараллеливания. Оценки эффективности распараллеливания построенных алгоритмов.
4.2	Генетические алгоритмы	Основные понятия и методы, лежащие в основе генетических алгоритмов поиска глобального оптимума. Этапы генетического алгоритма поиска оптимума, подходы к распараллеливанию алгоритма, распараллеливание алгоритма для систем с общей и распределенной памятью. Оценки эффективности распараллеливания построенных алгоритмов.

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

1. Афанасьев, К.Е. Основы высокопроизводительных вычислений. Т.2. Технологии параллельного программирования: учебное пособие: учебное пособие [Текст] / К.Е. Афанасьев, С.В. Стуколов, В.В. Малышенко, С.Н. Карабцев, Н.Е. Андреев; Кемеровский государственный университет. - Кемерово, 2013. – 434 с.
2. Афанасьев, К.Е. Основы высокопроизводительных вычислений. Т.3. Параллельные вычислительные алгоритмы: учебное пособие [Текст] / К.Е. Афанасьев, И.В. Григорьева, Т.С. Рейн; Кемеровский государственный университет. – Кемерово, 2012. – 185 с.
3. Белова И.М., Рассказов А.А. Параллельное программирование // И.М. Белова, А.А. Рассказов, МГИУ (Московский государственный индустриальный университет), 2012 - 101 стр. (http://e.lanbook.com/books/element.php?pl1_id=51752)
4. Богачёв К.Ю. Основы параллельного программирования: учебное пособие // К.Ю. Богачёв, "Бином. Лаборатория знаний", 2013 - 342 с. (http://e.lanbook.com/books/element.php?pl1_id=42626)

5. Гергель В.П. Теория и практика параллельных вычислений / Режим доступа: <http://www.intuit.ru/department/calculate/paralltp/>, свободный.
6. Гергель В. П., Высокопроизводительные вычисления для многопроцессорных многоядерных систем: учебник для вузов / Гергель В. П. - М. : Изд-во Московского университета, 2010. – 543с
7. Уильямс Э. Параллельное программирование на C++ в действии. Практика разработки многопоточных программ // Э. Уильямс, "ДМК Пресс", 2012 – 672 с. (http://e.lanbook.com/books/element.php?pl1_id=4813)

6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю)

6.1. Паспорт фонда оценочных средств по дисциплине (модулю)

№ п/п	Контролируемые разделы (темы) дисциплины (результаты по разделам)	Код контролируемой компетенции (или её части) / и её формулировка – по желанию	наименование оценочного средства
1.	Анализ параллельных алгоритмов. Принципы разработки параллельных методов.	ОПК-1 знать общенаучные базовые знания в области параллельных алгоритмов; уметь демонстрировать эти знания; владеть пониманием основных фактов, концепций, принципов построения параллельных алгоритмов; ПК-2 уметь понимать, совершенствовать и применять математические основы параллельных алгоритмов; ПК-11 знать основы параллельных алгоритмов; ПК-12 уметь применять на практике современные методы презентации материалов по параллельным алгоритмам;	контрольная работа
2.	Распараллеливание матрично-векторных операций.	ОПК-1 знать общенаучные базовые знания в области параллельных алгоритмов; уметь демонстрировать эти знания; владеть пониманием основных фактов,	отчет по лабораторной работе

№ п/п	Контролируемые разделы (темы) дисциплины (результаты по разделам)	Код контролируемой компетенции (или её части) / и её формулировка – по желанию	наименование оценочного средства
		концепций, принципов построения параллельных алгоритмов; ПК-2 уметь понимать, совершенствовать и применять математические основы параллельных алгоритмов;	
3.	Решение систем линейных алгебраических уравнений	ОПК-1 знать общенаучные базовые знания в области параллельных алгоритмов; уметь демонстрировать эти знания; владеть пониманием основных фактов, концепций, принципов построения параллельных алгоритмов; ПК-2 уметь понимать, совершенствовать и применять математические основы параллельных алгоритмов;	отчет по лабораторной работе
4.	Параллельные сортировки и генетические алгоритмы	ОПК-1 знать общенаучные базовые знания в области параллельных алгоритмов; уметь демонстрировать эти знания; владеть пониманием основных фактов, концепций, принципов построения параллельных алгоритмов; ПК-2 уметь понимать, совершенствовать и применять математические основы параллельных алгоритмов;	отчет по лабораторной работе

6.2. Типовые контрольные задания или иные материалы

6.2.1. Вопросы к зачету

1. Понятия вычислительная сложность и относительного времени выполнения

алгоритма.

2. Модель алгоритма, определение и примеры графа «операции - операнды» и расписания параллельного алгоритма.
3. Время выполнения, ускорение и эффективность параллельного алгоритма.
4. Понятия степени параллелизма компьютера и алгоритма, понятие паракомпьютера.
5. Понятие сверхлинейного ускорения и причины его возникновения.
6. Понятие средняя степень параллелизма алгоритма. Максимальное достижимое ускорение, закон Амдаля.
7. Этапы проектирования параллельного алгоритма.
8. Действия на этапах декомпозиции, проектирования коммуникаций, масштабирования подзадач, планирования вычислений.
9. Параллельные алгоритм умножения матрицы на вектор и матрицы на матрицу для систем с общей памятью.
10. Параллельный алгоритм умножения матрицы на вектор и матрицы на матрицу для систем с распределенной памятью.
11. Альтернативные параллельные алгоритмы матрично-векторного умножения: алгоритмы Виноградова и Фокса.
12. Теоретические оценки эффективности и ускорения алгоритмов умножения матриц.
13. Прямой и обратный проходы последовательного алгоритма метода Гаусса.
14. Идея параллельной реализации метода Гаусса для систем с общей памятью.
15. Способы декомпозиции матрицы СЛАУ при параллельной реализации метода Гаусса.
16. Параллельная реализация выбора ведущего элемента в методе Гаусса для систем с распределенной памятью.
17. Схема программной реализации параллельного варианта метода Гаусса с использованием MPI.
18. Теоретические оценки эффективности и ускорения параллельного алгоритма метода Гаусса.
19. Основные шаги последовательного алгоритма метода сопряженных градиентов. Предобуславливание матрицы системы линейных уравнений. Вычислительная трудоемкость алгоритма.
20. Формат CSIR хранения матрицы СЛАУ.
21. Распараллеливание метода сопряженных градиентов для систем с общей распределенной памятью. Ускорение и эффективность параллельного алгоритма метода сопряженных градиентов.
22. Последовательный алгоритм метода минимальных невязок, вычислительная трудоемкость метода минимальных невязок.
23. Распараллеливание метода минимальных невязок. Ускорение и эффективность параллельного алгоритма метода минимальных невязок.
24. Пузырьковая сортировка и ее чет-нечетная модификация. Вычислительная трудоемкость алгоритма чет-нечетной сортировки.
25. Распараллеливание алгоритма чет-нечетной перестановки для систем с общей распределенной памятью. Ускорение и эффективность параллельного алгоритма чет-нечетной сортировки.
26. Последовательная сортировка Шелла, вычислительная трудоемкость алгоритма сортировка Шелла.
27. Распараллеливание сортировки Шелла для систем с общей распределенной памятью. Ускорение и эффективность параллельного алгоритма чет-нечетной сортировки..
28. Последовательная быстрая сортировка, вычислительная трудоемкость алгоритма сортировка Шелла.
29. Распараллеливание быстрой сортировки для систем с общей распределенной памятью. Ускорение и эффективность параллельного алгоритма быстрой сортировки.
30. Принципы построения генетических алгоритмов на примере задачи о ферзях. Вычислительная трудоемкость генетических алгоритмов.

31. Подходы к распараллеливанию генетического алгоритма. Теоретические оценки ускорения и эффективности распараллеливания генетических алгоритмов.

1. критерии оценивания компетенций (результатов)

Курс ориентирован на формирование компетенций, направленных на приобретение общенаучных базовых знаний в области высокопроизводительных вычислений (ПК-1) и овладение способностью понимать и применять в исследовательской и прикладной деятельности теоретические положения и практические навыки реализации параллельных алгоритмов (ПК-3). Особое внимание в курсе уделяется приобретению практических навыков по реализации и анализу параллельных алгоритмов.

Критерием овладения теоретическим материалом по теоретическому анализу параллельных алгоритмов является контрольная работа по первой главе (знать).

Реализация программ из последующих разделов, представление отчетов по лабораторным работам подтверждает приобретение практических компонентов компетенций (уметь владеть).

Критерии оценивания теоретических знаний

Раздел 1 Анализ параллельных алгоритмов. Принципы разработки параллельных методов.

на отметку "зачтено"

- a. Различные определения алгоритма,
- b. требованиям, предъявляемым к алгоритмам.
- c. Критерии качества программного обеспечения
- d. Модель параллельного алгоритма в виде графа "операнд-операция" и расписания
- e. время выполнения параллельного алгоритма, ускорение и эффективность.
- f. Формальная модель ускорения.

Раздел 2 Распараллеливание матрично-векторных операций. Решение систем линейных алгебраических уравнений.

на отметку "зачтено"

- 1) Различные подходы к распределению элементов матрицы и вектора между процессорами вычислительной системы согласно выбранным базовым операциям при распараллеливании. Теоретические оценки ускорения и эффективности распараллеливания каждого алгоритмов.
- 2) Различные подходы к распределению элементов матриц между процессорами вычислительной системы согласно выбранным базовым операциям при распараллеливании. Теоретические оценки ускорения и эффективности распараллеливания каждого алгоритмов.

Раздел 3 Параллельные сортировки и генетические алгоритмы

на отметку "зачтено"

- 1) Параллельные алгоритм чет-нечетной сортировки. Оценки эффективности распараллеливания.
- 2) Параллельные алгоритм сортировки Шелла. Оценки эффективности распараллеливания.
- 3) Основные понятия и методы, лежащие в основе генетических алгоритмов поиска глобального оптимума.

Раздел 4 Параллельные алгоритмы на графах

на отметку "зачтено"

- 1) Подходы к распараллеливанию алгоритмов поиска кратчайших путей. Оценки эффективности параллельных алгоритмов.
- 2) Подходы к распараллеливанию алгоритмов оптимального разделения графов. Оценки эффективности параллельных алгоритмов

2. описание шкалы оценивания

Работы по каждому разделу оцениваются дифференцированно согласно формуле, (**Оценка-2**)^{*}8, где **Оценка** – оценка по пятибалльной шкале.

Максимальная сумма баллов, набираемая студентом по оценочным средствам – 96 баллов. Также студент может набрать баллы регулярно посещая занятия (0.5 балла за каждые 2 занятия (лекция и практика)) и активно работая на них (0.5 балла). В зависимости от суммарного количества набранных баллов в течение семестра, студенту выставляются следующие оценки: 0-59 баллов – «не зачленено»; 60-100 и более баллов – «зачленено».

6.2.2. Контрольная работа №1. Анализ параллельных алгоритмов. Принципы разработки параллельных методов

Вариант 1

32. Сложностные классы задач.
33. Понятие средней степени параллелизма алгоритма.
34. Модель алгоритма, определение и примеры графа «операции - операнды» и расписания параллельного алгоритма.

Вариант 2

1. Временная и пространственная сложность алгоритма.
2. Максимальное достижимое ускорение, закон Амдаля.
3. Понятие сверхлинейного ускорения и причины его возникновения.

Вариант 3

1. Модель алгоритма, определение и примеры графа «операции - операнды» и расписания параллельного алгоритма.
2. Понятие сверхлинейного ускорения и причины его возникновения.
3. Понятия степени параллелизма компьютера и алгоритма, понятие паракомпьютера. Декомпозиция вычислений.

Вариант 4

1. Время выполнения, ускорение и эффективность параллельного алгоритма.
2. Формальная модель ускорение.
3. Понятие сверхлинейного ускорения и причины его возникновения.

Вариант 5

1. Декомпозиция вычислений и проектирование коммуникаций.
2. Понятие средней степени параллелизма алгоритма.
3. Критерии качества программного обеспечения

Шкала оценивания: оценка «отлично» выставляется обучающемуся, если:

- даны ответы на все вопросы;
- ответы достаточно подробные, приводятся все необходимые определения, определения точны;
- приводятся примеры;
- при изложении вопроса студент владеет терминологией, все термины используются правильно;
- ответ на вопрос не содержит грамматических и стилистических ошибок;

- вычисления в примерах выполнены без ошибок.

оценка «хорошо» :

- дан ответ на два вопроса;
- приводятся все необходимые определения, возможно с некоторыми неточностями;
- при изложении вопроса студент владеет терминологией;
- приводятся примеры, хотя бы по одному вопросу.

оценка «удовлетворительно»:

- дан ответ на один вопрос;
- приводятся определения, возможно с некоторыми неточностями.

6.2.3. Лабораторные работы

2. Распараллеливание матрично-векторных операций.

Задание на лабораторную работу

1. Реализуйте последовательные и параллельные алгоритмы, предложенные в варианте (указывается только параллельный алгоритм). Убедитесь в правильности реализации.
2. Выполните серию расчетов для N от 1000 до 50000 с шагом 2500 (1000, 2500, 5000 и т. д.), выполните замеры времени, вычислите ускорение и эффективность.
3. Проанализируйте показатели производительности полученной параллельной реализации генетического алгоритма. Результаты опишите в отчете.
4. Оформить отчет по выполненной работе

Содержание отчета:

1. Титульный лист (образец на unesco.kemsu.ru)
2. Содержание
3. Цели и задачи
4. Описание работы
5. Анализ результатов и выводы
6. Список литературы

При анализе результатов замеров необходимо:

- 1) Указать на каком вычислительном ресурсе производятся замеры.
- 2) Привести таблицу и графики времени выполнения $T_p(n)$ для количества процессоров p от 1 до P (возможно с некоторым шагом, например, $p=1,2,4,\dots,P$), где P - максимальное количество процессоров, доступное на данном ресурсе.
- 3) Также для показательных случаев (для которых имеют место показатели эффективности $Sp > 0.5 * T_1/p$) привести график $T_n(p)$.
- 4) Таблицы и графики ускорения и эффективности в $Sp(n)$ и $Ep(n)$ для того же количества процессоров. В отчете непоказательные графики можно опустить, но таблицы необходимо привести.
- 5) В выводах необходимо отразить следующие пункты:
 1. Достигается ли в построенных параллельных реализациях выигрыш по времени по сравнению с последовательными вычислениями, если да, то для каких n и p .
 2. Имеет ли место монотонный рост ускорения и эффективности с увеличением p , для каких n и почему.
 3. Соответствуют ли полученные характеристики ускорения и эффективности теоретическим оценкам. Если не соответствуют, то чем объясняется такое несоответствие.
 4. Какой из реализованных методов показал себя лучше. При ответе на этот вопрос можно воспользоваться следующими критериями:
 1. Продемонстрировал стабильно лучшие (например, с ростом n) показатели производительности.

2. Позволил выполнить расчет для большего n.
3. Был проще в реализации.

Варианты 1. Умножение матрицы на вектор, три способа декомпозиции.

Вариант 2. Умножение матриц: первый способ, алгоритм Фокса.

Вариант 3. Умножение матриц: второй способ, алгоритм Кенона.

3. Решение систем линейных алгебраических уравнений

Задание на лабораторную работу

1. Реализуйте последовательные и параллельные алгоритмы, предложенные в варианте (указывается только параллельный алгоритм). Убедитесь в правильности реализации.
2. Выполните серию расчетов для N от 1000 до 50000 с шагом 2500 (1000, 2500, 5000 и т. д.), выполните замеры времени, вычислите ускорение и эффективность.
3. Проанализируйте показатели производительности полученной параллельной реализации генетического алгоритма. Результаты опишите в отчете.
4. Оформить отчет по выполненной работе

Содержание отчета:

7. Титульный лист (образец на unesco.kemsu.ru)
8. Содержание
9. Цели и задачи
10. Описание работы
11. Анализ результатов и выводы
12. Список литературы

При анализе результатов замеров необходимо:

- 6) Указать на каком вычислительном ресурсе производятся замеры.
- 7) Привести таблицу и графики времени выполнения $T_p(n)$ для количества процессоров p от 1 до P (возможно с некоторым шагом, например, $p=1,2,4,\dots,P$), где P - максимальное количество процессоров, доступное на данном ресурсе.
- 8) Также для показательных случаев (для которых имеют место показатели эффективности $S_p > 0.5 * T_1/p$) привести график $T_n(p)$.
- 9) 4) Таблицы и графики ускорения и эффективности в $S_p(n)$ и $E_p(n)$ для того же количества процессоров. В отчете непоказательные графики можно опустить, но таблицы необходимо привести.
- 10) В выводах необходимо отразить следующие пункты:
 1. Достигается ли в построенных параллельных реализациях выигрыш по времени по сравнению с последовательными вычислениями, если да, то для каких n и p .
 2. Имеет ли место монотонный рост ускорения и эффективности с увеличением p , для каких n и почему.
 3. Соответствуют ли полученные характеристики ускорения и эффективности теоретическим оценкам. Если не соответствуют, то чем объясняется такое несоответствие.
 4. Какой из реализованных методов показал себя лучше. При ответе на этот вопрос можно воспользоваться следующими критериями:
 1. Продемонстрировал стабильно лучшие (например, с ростом n) показатели производительности.
 2. Позволил выполнить расчет для большего n .
 3. Был проще в реализации.

Вариант 1. Метод Гаусса с выбором главного элемента по строке.

Вариант 2. Метод Гаусса с выбором главного элемента по столбцу.

Вариант 3. Метод Гаусса с выбором главного элемента по всей матрице.

4. Параллельные сортировки и генетические алгоритмы

Задание на лабораторную работу

1. Реализовать последовательные и параллельные алгоритмы сортировок, предложенные в варианте (указывается только параллельный алгоритм, в качестве последовательного алгоритма необходимо использовать быструю сортировку). Убедитесь в правильности реализации.

2. Выполните серию расчетов для N от 500 до 5000 с шагом 500, выполните замеры времени, вычислите ускорение и эффективность.

3. Результаты сравнения показателей эффективности сортировок опишите в отчете.

4. Реализуйте последовательный и параллельный генетический алгоритмы для задачи об N ферзях с использование реализованного алгоритма сортировки. Алгоритм сортировки необходимо использовать для упорядочивания особей в популяции по возрастанию количества угроз на доске.

5. Проанализируйте показатели производительности полученной параллельной реализации генетического алгоритма. Результаты опишите в отчете.

3. Оформить отчет по выполненной работе

Содержание отчета:

13. Титульный лист (образец на unesco.kemsu.ru)
14. Содержание
15. Цели и задачи
16. Описание работы
17. Анализ результатов и выводы
18. Список литературы

Для анализа результатов замеров необходимо:

11) Указать на каком вычислительном ресурсе производятся замеры.

12) Привести таблицу и графики времени выполнения $T_p(n)$ для количества процессоров p от 1 до P (возможно с некоторым шагом, например, $p=1,2,4,\dots,P$), где P - максимальное количество процессоров, доступное на данном ресурсе.

13) Также для показательных случаев (для которых имеют место показатели эффективности $S_p > 2^*T_1/p$) привести график $T_p(n)$.

14) Таблицы и графики ускорения и эффективности в $S_p(n)$ и $E_p(n)$ для того же количества процессоров. В отчете непоказательные графики можно опустить, но таблицы необходимо привести.

15) В выводах необходимо отразить следующие пункты:

1. Достигается ли в построенных параллельных реализациях выигрыш по времени по сравнению с последовательными вычислениями, если да, то для каких n и p .

2. Имеет ли место монотонный рост ускорения и эффективности с увеличением p , для каких n и почему.\

3. Соответствуют ли полученные характеристики ускорения и эффективности теоретическим оценкам. Если не соответствуют, то чем объясняется такое несоответствие.

4. Какой из реализованных методов показал себя лучше. При ответе на этот вопрос можно воспользоваться следующими критериями:

1. Продемонстрировал стабильно лучшие (или, например, с ростом n) показатели производительности.

2. Позволил выполнить расчет для большего n .

3. Был проще в реализации.

Вариант 1. Чет-нечетная сортировка.

Вариант 2. Параллельная сортировка Шелла.

Шкала оценивания: оценка «отлично» выставляется обучающемуся, если:

- Отчет правильно оформлен, структура отчета выдержанна;
- Описание работы достаточно полное;
- Представлен достаточно полный анализ результатов. Выводы соответствуют результатам представленных замеров.
- При изложении вопроса студент владеет терминологией, все термины используются правильно;
- Отчет не содержит грамматических и стилистических ошибок, рисунки и таблицы подписаны, список литературы содержит не менее 6 источников, оформлен согласно ГОСТ.

оценка «хорошо» :

- Отчет правильно оформлен, структура отчета выдержанна;
- Описание работы достаточно полное;
- Представлен достаточно полный анализ результатов. Выводы соответствуют результатам представленных замеров.

оценка «удовлетворительно»:

- Отчет правильно оформлен, структура отчета выдержанна;
- Описание работы присутствует, возможно недостаточно полное;
- Представлен анализ результатов. Выводы соответствуют результатам представленных замеров, возможно с некоторыми замечаниями.

7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

а) основная учебная литература:

1. Афанасьев, К.Е. Основы высокопроизводительных вычислений. Т.2. Технологии параллельного программирования: учебное пособие: учебное пособие [Текст] / К.Е. Афанасьев, С.В. Стуколов, В.В. Малышенко, С.Н. Карабцев, Н.Е. Андреев; Кемеровский государственный университет. - Кемерово, 2013. – 434 с.
2. Афанасьев, К.Е. Основы высокопроизводительных вычислений. Т.3. Параллельные вычислительные алгоритмы: учебное пособие [Текст] / К.Е. Афанасьев, И.В. Григорьева, Т.С. Рейн; Кемеровский государственный университет. – Кемерово, 2012. – 185 с.
3. Федотов И.Е. Модели параллельного программирования // И.Е. Федотов, Издательство "СОЛОН-Пресс", 2012, 384 с.
(http://e.lanbook.com/books/element.php?pl1_id=13807)

б) дополнительная учебная литература:

1. Белова И.М., Рассказов А.А. Параллельное программирование // И.М. Белова, А.А. Рассказов, МГИУ (Московский государственный индустриальный университет), 2012 - 101 стр.
(http://e.lanbook.com/books/element.php?pl1_id=51752)
2. Богачёв К.Ю. Основы параллельного программирования: учебное пособие // К.Ю. Богачёв, "Бином. Лаборатория знаний", 2013 - 342 с.
(http://e.lanbook.com/books/element.php?pl1_id=42626)

3. Гергель В.П. Теория и практика параллельных вычислений / Режим доступа: <http://www.intuit.ru/department/calculate/paralltp/>, свободный.
4. Гергель В. П., Высокопроизводительные вычисления для многопроцессорных многоядерных систем: учебник для вузов / Гергель В. П. - М. : Изд-во Московского университета, 2010. – 543с
5. Уильямс Э. Параллельное программирование на C++ в действии. Практика разработки многопоточных программ // Э. Уильямс, "ДМК Пресс", 2012 – 672 с. (http://e.lanbook.com/books/element.php?pl1_id=4813)
6. Tutorial on MPI: The Message-Passing Interface William Gropp Mathematics and Computer Science Division Argonne National Laboratory Argonne, IL 60439 www-unix.mcs.anl.gov/mpi/tutorial

8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее – сеть «Интернет»), необходимых для освоения дисциплины (модуля)

ЭУМК «Виртуальный лабораторный практикум по параллельным вычислениям»

[Электронный ресурс] / Режим доступа: virtlab.kemsu.ru, после регистрации

www.mathnet.ru – общероссийский математический портал;

http://e.lanbook.com/books/?p_f_1_temp_id=18&p_f_1_65=917&p_f_1_63=&p_f_1_67= – электронно-библиотечная система, издательство «Лань»;

www.elibrary.ru – научная электронная библиотека;

www.lib.mexmat.ru – электронная библиотека механико-математического факультета МГУ;

http://www.newlibrary.ru/genre/nauka/matematika/kompyutery_i_matematika/ – электронная библиотека по математике;

http://www.edu.ru/modules.php?op=modload&name=Web_Links&file=index&l_op=viewlink&id=2720 – федеральный портал российского профессионального образования: Математика и естественно-научное образование;

9. Методические указания для обучающихся по освоению дисциплины (модуля)

К современному специалисту общество предъявляет достаточно широкий перечень требований, среди которых немаловажное значение имеет наличие у выпускников определенных способностей и умения самостоятельно добывать знания из различных источников, систематизировать полученную информацию. Формирование такого умения происходит в течение всего периода обучения через участие студентов в практических занятиях, выполнение контрольных заданий и тестов, написание курсовых и выпускных квалификационных работ. При этом самостоятельная работа студентов играет решающую роль в ходе всего учебного процесса.

9.1. Советы по планированию и организации времени, необходимого для изучения дисциплины. Рекомендуется следующим образом организовать время, необходимое для изучения дисциплины:

Данный курс предполагает реализацию довольно большого объема программного кода, а так же разбор кода, представленного в материале курса. Данный курс предусматривает использование кластеров Центра коллективного пользования КемГУ hphead.kemsu.ru, master.kemsu.ru доступного только из сети университета. Поэтому тестирование кода с замером временных характеристик, может проводиться только при выполнении лабораторных работ в компьютерных классах или с использованием виртуальной лаборатории virtlab.kemsu.ru.

Практически все темы курса, за исключением первой, автономны и в случае самостоятельной работы могут изучаться в любом порядке.

Перед реализацией программ предполагается изучение теоретического материала раздела 1. Для каждой последующей темы необходимо изучить теоретический материала. Для каждой задачи по рассматриваемой теме строятся теоретические оценки производительности, реализуется программа, проводится тестирование правильности работы программы, после чего замеряются временные характеристики реализованного алгоритма и производится сравнение с теоретическими оценками. По результатам этой работы составляется отчет по текущей теме.

Отчеты по лабораторным работам, а также результаты итогового тестирования являются ключевыми моментами при получении зачета по дисциплине.

9.1 Указания по оформлению отчета по лабораторной работе

Структура отчета:

1) Титульный лист

2) Цели и задачи работы. В целях следует указать на освоение какого класса алгоритмов направлена данная лабораторная работа. Целью всех лабораторных работ являются: построение теоретических оценок производительности алгоритмов, построение реализаций алгоритмов с использование OpenMP, MPI, а также эффективных гибридных реализаций, анализ соответствия построенных программных реализаций теоретическим оценкам эффективности. При описании задач следует указать задачи, выполнение которых предусмотрено по данной теме.

3) Результаты работы. Необходимо привести теоретические оценки производительности. Описание программ, сами программы приводятся в приложении. Основой анализа являются таблицы и графики времени выполнения, ускорения и эффективности.

4) Заключение. В этом пункте необходимо сформулировать выводы по результатам, приведенным в предыдущем пункте. Для этого следует последовательно ответить на вопросы.

1. Соответствуют ли экспериментальные оценки ускорения и эффективности теоретическим?
2. Удалось ли получить устойчивый рост ускорения и эффективности с ростом n ?
3. Удалось ли обеспечить разномерную загрузку вычислительных устройств?
4. Удалось ли обеспечить минимальный обмен данными, соответствует ли объем пересылаемых данных объему вычислений?
5. Удалось ли построить эффективные гибридные реализации?
6. Является построенная программная реализация масштабируемой?
7. Если одна и та же задача решалась несколькими способами, то необходимо дать обоснованное заключение о том, какой способ оказался более эффективным.

5) Список литературы, оформляется в соответствии с ГОСТ-ом.

10. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Компьютерные классы с набором лицензионного базового программного обеспечения для проведения лабораторных занятий и доступом по ftp или ssh к серверам Центра коллективного пользования высокопроизводительными ресурсами КемГУ hpchead.kemsu.ru, master.kemsu.ru. ЭУМК «Виртуальный лабораторный практикум по параллельным вычислениям» [Электронный ресурс] / Режим доступа: virtlab.kemsu.ru, после регистрации. Для лабораторных работ и консультаций Система поддержки учебного процесса iais.kemsu.ru.

11. Иные сведения и материалы

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

Образование обучающихся с ограниченными возможностями здоровья может быть организовано по выбору студента в следующем виде:

- совместно с другими обучающимися: студент посещает занятия на общих основаниях и непосредственно вовлекается в учебный процесс;
- дистанционно посредством телекоммуникационных технологий: студент прослушивает материал занятий в режиме реального времени, по средствам прямого телемоста (применение Skype или других аналогичных программ и технологий), не находясь непосредственно в учебной аудитории;
- в индивидуальном порядке: преподаватель занимается со студентом индивидуально контактно или посредством телекоммуникационных технологий.

По окончании изучения курса со студентом проводится индивидуальное собеседование, на котором он демонстрирует полученные знания. В случае необходимости, студенту может заранее быть выдано индивидуальное практическое задание, для самостоятельной подготовки (за месяц или за две недели).

Для инвалидов по слуху предусмотрены следующие особенности проведения учебного процесса:

1. Преподаватель предоставляет студенту учебно-методические материалы, необходимые для освоения изучаемого материала (программа курса, план занятия, опорный конспект, методические пособия или слайд презентации, в случае наличия).
2. Лекционный материал преподается в наглядном виде слайд презентаций или сопровождается схемами, наглядными таблицами.
3. Вместо устного ответа студентам предлагается отвечать письменно.
4. Предусматриваются индивидуальные консультации со студентом, на котором может присутствовать сурдопереводчик (университет не обязуется предоставлять сурдопереводчика).

Для инвалидов по зрению предусмотрены следующие особенности проведения учебного процесса:

1. Предусматриваются индивидуальные консультации со студентом, во время которых преподаватель в медленном спокойном темпе объясняет учебный материал (возможно повторно), заостряя внимание на ключевых понятиях.
2. Вместо письменного ответа студентам предлагается отвечать устно.
3. Предлагается ознакомиться с литературой по курсу, написанной шрифтом Брайля, при наличии.

Для инвалидов опорно-двигательного аппарата предусмотрены следующие особенности проведения учебного процесса:

1. Преподаватель предоставляет студенту учебно-методические материалы, необходимые для освоения изучаемого материала (программа курса, план занятия,

- опорный конспект, методические пособия или слайд презентации, в случае наличия).
2. Лекционный материал преподается в наглядном виде слайд презентаций или сопровождается схемами, наглядными таблицами.
 3. Вместо письменного ответа студентам предлагается отвечать устно.
 4. Предусматриваются индивидуальные консультации со студентом.

Составитель: Григорьева И.В. доцент кафедры ЮНЕСКО по информационно-вычислительным технологиям КемГУ
