

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ

Проректор по учебной работе и
образовательным инновациям

О.И. Чуприс

2019 г.

Регистрационный № УД- 7126 /уч.

Основы параллельных вычислений

Учебная программа учреждения высшего образования
по учебной дисциплине для специальностей:

- | | |
|---------------|--|
| 1-31 03 03 | Прикладная математика (по направлениям)
направления специальности |
| 1-31 03 03-01 | Прикладная математика (научно-производственная
деятельность) |

2019 г.

Учебная программа составлена на основе образовательного стандарта высшего образования ОСВО 1-31 03 03-2013 и учебного плана УВО № G31-173/уч. от 30/05/2013.

СОСТАВИТЕЛЬ:

Н.А. Лиходед, профессор кафедры вычислительной математики Белорусского государственного университета, доктор физико-математических наук, профессор

РЕЦЕНЗЕНТЫ:

В.Б. Малютин, ведущий научный сотрудник Института математики НАН, доктор физико-математических наук, профессор;

А.О. Сикорский – директор ООО «ЯндексБел».

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой вычислительной математики (протокол № 12 от 09.04.2019);

Научно-методическим Советом БГУ (протокол № 4 от 22.04.2019)

Заведующий кафедрой
вычислительной математики



В.И. Репников



ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Цели и задачи учебной дисциплины

Учебная дисциплина «Основы параллельных вычислений» знакомит студентов с подходами к решению задач отображения алгоритмов, задаваемых последовательными программами, на многоядерные персональные компьютеры, на параллельные компьютеры с распределенной памятью, на графические процессоры. Рассматриваются, в частности, следующие задачи: получение информационной структуры алгоритма, получение параллельных циклов, выбор зерна вычислений, организация параллельных и распределенных вычислительных процессов, распараллеливание алгоритмов для реализации на графических процессорах, построение параллельных алгоритмов поиска кратчайших путей в графе, улучшение локальности вычислений, аффинные преобразования гнезд циклов.

Цель учебной дисциплины – овладеть основами разработки параллельных алгоритмов, получить навыки использования сведений по методам и алгоритмам параллельных и распределенных вычислений, формирование составной части знаний, получаемых будущими специалистами в процессе учебы и необходимых им в дальнейшем для успешной работы, формирование у студентов основ математического мышления, изучение различных подходов к построению вычислительных алгоритмов, ориентированных на параллельные компьютеры.

Задачи учебной дисциплины:

1. Введение в проблематику статического распараллеливания, основанного на знании информационной структуры алгоритмов.
2. Изучение основных терминов и понятий, математического аппарата и моделей параллельных вычислений.
3. Получение теоретических и практических основ выявления параллелизма, распараллеливания алгоритмов, преобразования последовательных программ в параллельные, организации параллельных вычислительных процессов.

Место учебной дисциплины в системе подготовки специалиста с высшим образованием.

Учебная дисциплина относится к циклу дисциплин специализации компонента учреждения образования.

Программа составлена с учетом **межпредметных связей** с учебными дисциплинами. Основой для изучения учебной дисциплины являются следующие учебные дисциплины первой ступени высшего образования: «Вычислительные методы алгебры», «Методы численного анализа», «Алгоритмы и структуры данных».

Требования к компетенциям

Освоение учебной дисциплины «Основы параллельных вычислений» должно обеспечить формирование следующих академических, социально-личностных и профессиональных компетенций:

академические компетенции:

АК-1. Уметь применять базовые научно-теоретические знания для решения теоретических и практических задач.

АК-2. Владеть системным и сравнительным анализом.

АК-3. Владеть исследовательскими навыками.

АК-4. Уметь работать самостоятельно.

АК-5. Быть способным порождать новые идеи (обладать креативностью).

АК-6. Владеть междисциплинарным подходом при решении проблем.

АК-7. Иметь навыки, связанные с использованием технических устройств, управлением информацией и работой с компьютером.

социально-личностные компетенции:

СЛК-6. Уметь работать в команде.

профессиональные компетенции:

ПК-1. Работать с научно-технической, нормативно-справочной и специальной литературой.

В результате изучения дисциплины обучаемый должен:

знать:

- проблематику статического распараллеливания;
- основные термины и понятия, математический аппарат и модели параллельных вычислений;
- теоретические основы выявления параллелизма, распараллеливания алгоритмов, преобразования последовательных программ в параллельные, организации параллельных вычислительных процессов;

уметь:

- обнаруживать параллелизм;
- распределять операции и данные алгоритма между процессорами;
- устанавливать порядок выполнения операций и обмена данными.

владеет:

- основами разработки параллельных программ для компьютеров с общей и компьютеров с распределенной памятью;
- программными средствами для организации параллельных вычислительных процессов.

Структура учебной дисциплины

Дисциплина изучается в шестом семестре. Всего на изучение учебной дисциплины «Основы параллельных вычислений» для дневной формы получения образования отведено: 106 часов, в том числе 68 аудиторных часов, из них: лекции – 34 часов, лабораторные занятия – 30 часов, управляемой самостоятельной работы – 4 часа.

Трудоемкость учебной дисциплины составляет 3 зачетные единицы.

Форма текущей аттестации по учебной дисциплине – зачет.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Раздел 1. Введение. Реализация параллельных циклов на многоядерных компьютерах

Тема 1.1. Основные понятия параллельных вычислений

Цель и задача параллельных вычислений. Терминология параллельных вычислений.

Тема 1.2. Блочные матрицы. Блочные алгоритмы

Алгоритмы, представленные параллельными циклами. OpenMP-реализация параллельных точечных и блочных алгоритмов на многоядерных компьютерах.

Раздел 2. Информационная структура алгоритма

Тема 2.1. Анализ зависимостей

Функции, задающие индексы массивов данных. Зависимости. Представление зависимостей. Функции зависимостей. Графы зависимостей. Типы зависимостей. Способы устранения ложных зависимостей.

Тема 2.2. Инструментальные средства для получения информационной структуры алгоритма

Инструментальные средства для получения информационной структуры алгоритма. Визуализация информационной структуры алгоритма, получение информации о зависимостях, типах зависимостей, получение функций зависимостей и графов зависимостей с помощью инструментальных средств.

Раздел 3. Организация параллельных вычислений.

Тема 3.1. Тайлинг

Разбиение операций алгоритма на макрооперации-тайлы. Техника тайлинга. Допустимость тайлинга.

Тема 3.2. Организация параллельных вычислительных процессов. Параллельные последовательности зернистых вычислений

Задание зерна вычислений. Функции, задающие распределение зерен между процессорами. Условия параллельности последовательностей зернистых вычислений. Задачи (этапы) статического распараллеливания: получение информационной структуры алгоритма, выявление потенциального параллелизма алгоритма, выбор зерна вычислений, распределение входных данных, выделение массивов, согласование распределения операций и данных, улучшение локальности, организация обмена данными.

Разработка и реализация на суперкомпьютере параллельных алгоритмов для выполнения матричных операций, решения систем линейных алгебраических уравнений, решения дифференциальных уравнений в частных производных.

Тема 3.3. Распараллеливание алгоритмов для реализации на графических процессорах

Мультипроцессоры графического процессора. Виды памяти. Организация потоков и блоков потоков. Основные положения для распараллеливания алгоритмов при программировании общего назначения на графических процессорах. Разработка и реализация параллельных алгоритмов на графических процессорах.

Тема 3.4. Параллельные алгоритмы поиска в графе

Параллельные алгоритмы обработки графов. Параллельный алгоритм нахождения кратчайшего пути. Параллельный алгоритм поиска в ширину.

Раздел 4. Некоторые теоретические аспекты параллельных вычислений

Тема 4.1. Параллельная форма алгоритма. Концепция неограниченного параллелизма. Основные классы современных вычислительных систем

Параллельные множества операций, параллельная форма алгоритма, параллельные последовательности вычислений. Параллельные вычислительные свойства и параллельная структура алгоритма. Суть и основная цель концепции неограниченного параллелизма. Принцип сдваивания. Процесс рекуррентного сдваивания. Классификация Флинна параллельных компьютеров.

Тема 4.2. Характеристики параллельных вычислительных процессов и систем. Списки Top 500 и Graph 500

Обоснование определения понятий производительности, загруженности, ускорения вычислительной системы. Формулы для нахождения максимальных характеристик. Законы Амдала. Сетевой закон Амдала. Закон Густавсона-Барсиса. Сравнительный анализ законов Амдала и Густавсона-Барсиса. Список Top 500. Список Graph 500.

Раздел 5. Параллельная структура алгоритма

Тема 5.1. Распараллеливание, использующее уровни зависимостей

Распараллеливание, использующее уровни зависимостей. Алгоритм Аллена и Кеннеди распараллеливания гнезд тесно вложенных циклов.

Тема 5.2. Таймирующие функции

Таймирующие функции. Строгие таймирующие функции. Расщепляющие функции. Скошенный параллелизм. Векторные таймирующие функции.

Тема 5.3. Получение таймирующих функций для однородных гнезд циклов

Получение параметров таймирующих расщепляющих, строгих таймирующих функций посредством решения вспомогательных систем неравенств

и уравнений. Алгоритм получения таймирующих функций.

Тема 5.4. Аффинные преобразования гнезд вложенных циклов. Генерация кода

Преобразования вложенных циклов, задаваемые векторными таймирующими функциями. Условия параллельности внутренних циклов после преобразования. Условия параллельности внешних циклов после преобразования. Генерация кода после аффинного преобразования гнезда циклов. Генерация кода и OpenMP-реализация параллельных блочных алгоритмов на многоядерных компьютерах.

Тема 5.5. Обзор методов и инструментальные средства автоматизированного распараллеливания алгоритмов

Методы статического распараллеливания, основанные на многомерном таймировании, улучшения локальности и тайлинге. Системы ParLoc, ParProc, LooPo, CLooG, PLUTO.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ
Дневная форма получения образования

№ п/п	Название раздела, темы	Количество часов				Количество часов УСР	Форма контроля знаний
		Аудиторные					
		Лекции	Пр. и сем. зан.	Лаб. зан.	Иное		
1	Введение. Реализация параллельных циклов на многоядерных компьютерах	2		4			Коллоквиум
1.1	Основные понятия параллельных вычислений	1					Устный опрос
1.2	Блочные матрицы. Блочные алгоритмы	1		4			Отчет по лабораторной работе с ее устной защитой
2	Информационная структура алгоритма	6		5		1	Коллоквиум
2.1	Анализ зависимостей	5		4		1	Устный опрос. Отчет по лабораторной работе с ее устной защитой. Собеседование
2.2	Инструментальные средства для получения информационной структуры алгоритма	1		1			Отчет по лабораторной работе с ее устной защитой
3	Организация параллельных вычислений	10		14		2	Контрольная работа по темам 3.1, 3.2, коллоквиум
3.1	Тайлинг (разбиение множества операций алгоритма на макрооперации)	2		2			Устный опрос
3.2	Организация параллельных вычислительных процессов. Параллельные последовательности зернистых вычислений	4		6		2	Устный опрос. Отчет по лабораторной работе с ее устной защитой. Собеседование
3.3	Распараллеливание алгоритмов для реализации на графических процессорах (GPU)	2		4			Устный опрос
3.4	Параллельные алгоритмы	2		2			Устный опрос

	поиска в графе						
4	Некоторые теоретические аспекты параллельных вычислений	4					Коллоквиум
4.1	Параллельная форма алгоритма. Концепция неограниченного параллелизма. Основные классы современных вычислительных систем	2					Устный опрос
4.2	Характеристики параллельных вычислительных процессов и систем. Списки Top 500 и Graph 500	2					Устный опрос
5	Параллельная структура алгоритма	12		7		1	Коллоквиум
5.1	Распараллеливание, использующее уровни зависимостей	1					Устный опрос
5.2	Таймирующие функции	1					Устный опрос
5.3	Получение таймирующих функций для однородных гнезд циклов	2					Устный опрос
5.4	Аффинные преобразования гнезд вложенных циклов. Генерация кода	6		5		1	Устный опрос. Отчет по лабораторной работе с ее устной защитой. Собеседование
5.5	Обзор методов и инструментальных средств автоматизированного распараллеливания алгоритмов	2		2			Устный опрос
ИТОГО		34		30		4	

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Перечень основной литературы

1. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. – Санкт-Петербург. БХВ-Петербург. 2002. – 600 с.
2. Воеводин В. В. Вычислительная математика и структура алгоритмов. – Москва: Изд-во МГУ, 2006. – 112 с. <http://parallel.ru/info/parallel/voevodin/>
3. Лиходед Н.А. Методы распараллеливания гнезд циклов: Курс лекций. – Мн.: БГУ. 2008. – 100 с.
4. [Электрон. ресурс – \\fpmi-stud\Subfaculty\Каф. Выч. Мат\Параллельные вычисления\Лекции]

Перечень дополнительной литературы

5. Гервич Л.Р., Штейнберг Б.Я., Юрушкин М.В. Разработка параллельных программ с оптимизацией использования структуры памяти. – Ростов-на-Дону. Изд-во Южного федерального университета, 2014. – 120 с.
6. Антонов А.С. Параллельное программирование с использованием технологии OpenMP. – Москва, МГУ. 2009. – 77 с. http://parallel.ru/tech/tech_dev/OpenMP/examples
7. [Электрон. ресурс – \\fpmi-stud\Subfaculty\Каф. Выч. Мат\Параллельные вычисления\Инструментарий] Система Depend grapher.
8. LooPo. www.fmi.uni-passau.de/loopo.
9. Kim D., Rajopadhye S. Parameterized tiling for imperfectly nested loops. // Technical Report CS-09-101, Colorado State University, Department of Computer Science, February 2009.
10. Tavarageri S., Hartono A., Baskaran M., Pouchet L.-N., Ramanujam J., Sadayappan, P. Parametric tiling of affine loop nests // Proc. 15th Workshop on Compilers for Parallel Computers. Vienna, Austria, July 2010.
11. Лиходед Н.А., Толстикова А.А. Параллельные последовательности зернистых вычислений // Доклады НАН Беларуси. 2010. Т. 54, № 4. С. 36–41.
12. Антонов А.С. Параллельное программирование с использованием технологии MPI. – Москва, МГУ. 2004. – 71 с. <http://parallel.ru/info/parallel/antonov/>
13. Воеводин Вл.В., Воеводин Вад.В. Спасительная локальность суперкомпьютеров // Открытые системы. 2013, № 9. С. 12–15.
14. Боресков А.В., Харламов А.А. Основы работы с технологией CUDA. – Москва. ДМК Пресс, 2010. – 232 с.
15. Baskaran M., Ramanujam J., Sadayappan, P. Automatic C-to- CUDA code generation for affine programs // Proceedings of the Compiler Construction, 19th International Conference. Part of the Joint European Conferences on Theory and Practice of Software. Paphos, Cyprus, March 20–28, 2010.
16. Гегель В.П. Теория и практика параллельных вычислений. – Москва: Интернет-Университет, БИНОМ. Лаборатория знаний, 2007.
17. Yoo A., Chow E., Henderson K., McLendon W., Hendrickson B., Catalyurek U.V: A scalable distributed parallel breadth-first search algorithm on BlueGene/L // Proceedings of the ACM/IEEE Supercomputing 2005 Conference. November 2005.
18. Buluc A., Madduri K.: Parallel breadth-first search on distributed memory systems // Proceedings of the ACM/IEEE Supercomputing 2011 Conference. November 2011.
19. Электрон. ресурс – \\fpmi-stud\Subfaculty\Каф. Выч. Мат\Параллельные вычисления\Лекции] Система ParProc.
20. PLUTO: pluto-compiler.sourceforge.net.
21. CLooG: The Chunky Loop Generator. <http://www.cloog.org>.

Примерный перечень заданий для управляемой самостоятельной работы студентов

Тема «Информационная структура алгоритма».

1. Найти зависимости указанного алгоритма.
2. Найти функции зависимостей указанного алгоритма: вид функции, область определения.
3. Построить развернутый и редуцированный графы зависимостей указанного алгоритма.

Форма контроля – собеседование.

Тема «Организация параллельных вычислений».

1. Указать распределение входных и выходных данных одного из вариантов параллельного алгоритма прямого хода метода Гаусса.
2. Произвести выделение и приватизацию массивов одного из вариантов параллельного алгоритма перемножения матриц.
3. Записать коммуникационные операции одного из вариантов параллельного алгоритма перемножения матриц.

Форма контроля – собеседование.

Тема «Параллельная структура алгоритма. Генерация кода».

1. Обосновать параллельность внутреннего цикла после применения к алгоритму заданного аффинного преобразования.
2. Обосновать параллельность внешнего цикла после применения к алгоритму заданного аффинного преобразования.
3. Сгенерировать псевдокод, получаемый после заданного аффинного преобразования.

Форма контроля – собеседование.

Перечень рекомендуемых средств диагностики и методика формирования итоговой оценки

Для диагностики компетенций в рамках учебной дисциплины рекомендуется использовать следующие формы:

1. Устная форма: устный опрос, собеседование, коллоквиум.
2. Письменная форма: контрольные работы.
3. Устно-письменная форма: отчеты по лабораторным работам с их устной защитой.

Формой текущей аттестации по дисциплине «Основы параллельных вычислений» учебным планом предусмотрен экзамен.

При формировании итоговой оценки используется рейтинговая оценка знаний студента, дающая возможность проследить и оценить динамику процесса достижения целей обучения. Рейтинговая оценка предусматривает использование весовых коэффициентов для текущего контроля знаний студентов по дисциплине.

Примерные весовые коэффициенты, определяющие вклад текущего контроля знаний в рейтинговую оценку:

- работа на лабораторных занятиях – 50 %;

- контрольная работа – 30 %;
- коллоквиум – 20 %.

Рекомендуемая тематика контрольных работ

Контрольная работа. Организация параллельных вычислительных процессов.

Коллоквиум. Информационная структура алгоритма, параллельная структура алгоритма

Описание инновационных подходов и методов к преподаванию учебной дисциплины (эвристический, проективный, практико-ориентированный)

При организации образовательного процесса большинства практических занятий используется практико-ориентированный подход, который предполагает освоение содержания учебного материала через решение практических задач, а также приобретение навыков эффективного выполнения разных видов профессиональной деятельности.

Кроме этого, при организации образовательного процесса используется комбинация методов группового обучения, проектного обучения и учебной дискуссии. Комбинация методов предполагает: ориентацию на генерирование идей, приобретение навыков для решения исследовательских, творческих и коммуникационных задач, появление нового уровня понимания изучаемой темы, применение знаний (теорий, концепций) при решении проблем, определение способов их решения.

Методические рекомендации по организации самостоятельной работы обучающихся

Для организации самостоятельной работы студентов магистратуры по учебной дисциплине следует использовать современные информационные технологии: разместить в сетевом доступе комплекс учебных и учебно-методических материалов (учебно-программные материалы, учебное издание для теоретического изучения дисциплины, презентации лекций, методические указания к практическим занятиям, электронные версии домашних заданий, материалы текущего контроля и текущей аттестации, позволяющие определить соответствие учебной деятельности обучающихся требованиям образовательных стандартов высшего образования и учебно-программной документации, в том числе вопросы для подготовки к зачёту, задания, вопросы для самоконтроля, список рекомендуемой литературы, информационных ресурсов и др.).

Примерный перечень вопросов к зачету

1. Определение зависимости между операциями. Типы зависимостей (истинная зависимость, антизависимость, зависимость по выходу).
2. Условие параллельности внешних циклов после применения аффинного преобразования гнезда циклов.
3. Таймирующие функции, строгие таймирующие функции, расщепляющие функции (определения и примеры).
4. Распараллеливание, использующее уровни зависимостей (метод Аллена-Кеннеди).
5. Функция зависимостей. Вектор зависимости. Однородные и аффинные зависимости.
6. Улучшение локальности гнезд циклов при последовательных вычислениях. Пространственная локальность. Временная локальность.
7. Развернутый и редуцированный графы зависимостей.
8. Теорема о получении таймирующих функций для однородных гнезд циклов.
9. Разбиение множества операций алгоритма на макрооперации-тайлы. Примеры использования тайлинга.
10. Условия параллельности внутренних циклов после применения аффинного преобразования гнезда циклов.
11. Основные положения для распараллеливания алгоритмов при программировании общего назначения на графических процессорах.
12. Алгоритм получения таймирующих функций для однородных гнезд циклов.
13. Устранение ложных зависимостей путем переобозначения или увеличения размерности массивов.
14. Получение параллельных зернистых последовательностей вычислений (на примере).
15. Векторные таймирующие функции и аффинные преобразования циклов (определения и примеры).
16. Основные задачи статического распараллеливания алгоритмов, заданных последовательными программами.
17. Этапы и пример генерация кода после аффинного преобразования гнезда циклов.
18. Классификация Флинна параллельных компьютеров.
19. Этапы организации параллельных зернистых вычислительных процессов и обмена данными.
20. Производительность, загруженность, ускорение вычислительной системы (определения). Формулы, выражающие производительность и ускорение системы из s процессоров через загруженность.
21. Цель и задача параллельных вычислений. Терминология параллельных вычислений.
22. Параллельная форма алгоритма. Наименьшая возможная высота алгоритма, зависящего от N переменных.
23. Список Top 500. Список Graph 500.

24. Закон Густавсона-Барсиса.
25. Формулы максимальных производительности, загруженности, ускорения. Первый закон Амдала.
26. Основные факторы, определяющие скорость выполнения алгоритма на параллельном компьютере. Проблемы вычислительной математики, которые возникают при использовании параллельных компьютеров.
27. Перестановка циклов: достаточные условия, применение для распараллеливания и тайлинга.
28. Сетевой закон Амдала.
29. Параллельные множества операций, параллельная форма алгоритма, параллельные последовательности вычислений. Параллельные вычислительные свойства и параллельная структура алгоритма.
30. Второй закон Амдала.
31. Распределение циклов: достаточные условия, применение для распараллеливания и тайлинга.
32. Использование таймирующих функций.
33. Концепция неограниченного параллелизма. Максимальное ускорение системы из s ядер, реализующей N операций алгоритма высоты m .
34. Сравнительный анализ второго закона Амдала и закона Густавсона-Барсиса. «
35. Параллельные множества операций. Естественный параллелизм. Внутренний параллелизм. Параллельные вычислительные процессы.
36. Допустимость тайлинга (необходимые и достаточные условия, достаточные условия; формулировки и иллюстрации).
37. Сетевой аналог третьего закон Амдала.
38. Двухуровневый тайлинг для распараллеливания алгоритмов при программировании общего назначения на графических процессорах. Пример.

ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ

Название учебной дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу

**ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ ПО
ИЗУЧАЕМОЙ УЧЕБНОЙ ДИСЦИПЛИНЕ**

на ____ / ____ учебный год

№ п/п	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры
_____ (протокол № ____ от _____ 20__ г.)

Заведующий кафедрой
Доцент _____ В.И. Репников

УТВЕРЖДАЮ
Декан факультета
