

«УТВЕРЖДАЮ»

Проректор по научной работе

Драгунов В.К.

« 16 » июня 2015 г.



Программа аспирантуры

Направление 09.06.01 Информатика и вычислительная техника

Направленность (специальность) 05.13.11 Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей

РАБОЧАЯ ПРОГРАММА

дисциплины по выбору

«Организация распределенных и параллельных систем»

Индекс дисциплины по учебному плану: Б1.В.ДВ.4.1

Всего: 108 часов

Семестр 7, в том числе

6 часов – контактная работа,
84 часа – самостоятельная работа,
18 часов – контроль

Программа составлена на основе федерального государственного образовательного стандарта высшего образования (уровень подготовки кадров высшей квалификации) по направлению подготовки 09.06.01 Информатика и вычислительная техника, утвержденного приказом Минобрнауки России от 30 июля 2014 г. № 875, и паспорта специальности, указанной в номенклатуре специальностей научных работников 05.13.11 Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей, утвержденной приказом Минобрнауки России от 25 февраля 2009 г. № 59.

ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью изучения дисциплины является освоение организации компьютерных систем, их технических характеристик и программного обеспечения, методов разработки и оптимизации параллельных программ.

Задачами дисциплины являются:

- изучение методов организации сильно связанных и распределенных компьютерных систем и их технических характеристик;
- освоение языков параллельного программирования: MPI, MULTITHREADING, функциональных языков, языков крупноблочного структурного программирования;
- освоение методов и технологий разработки больших параллельных программ и их оптимизации;
- изучение моделей параллельных процессов, методов и алгоритмов управления процессами выполнения параллельных программ на компьютерных системах.

В процессе освоения дисциплины **формируются следующие компетенции:**

- способность проектировать и осуществлять комплексные исследования, в том числе междисциплинарные, на основе целостного

системного научного мировоззрения с использованием знаний в области истории и философии науки (УК-2);

– владение методологией теоретических и экспериментальных исследований в области профессиональной деятельности (ОПК-1);

– способность к разработке новых методов исследования и их применению в самостоятельной научно-исследовательской деятельности в области профессиональной деятельности (ОПК-3);

– владение методами создания программ и программных систем для параллельной и распределенной обработки данных, языками и инструментальными средствами параллельного программирования (ПК-3);

– способность разрабатывать и исследовать математические модели объектов, систем, процессов и технологий, предназначенных для проведения расчетов, анализа, подготовки решений (ПК-6);

– владение методами и средствами виртуализации и виртуального отображения реальности в современных распределенных системах (ПК-9).

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБРАЗОВАНИЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ

В результате освоения дисциплины обучающийся должен демонстрировать **следующие результаты образования:**

знать:

– архитектурную организацию современных компьютеров и КС (УК-2);

– методы, технологии и языки параллельного программирования (ПК-3);

– методы и операционные средства управления параллельными процессами (ОПК-3);

уметь:

– разрабатывать параллельные программы решения задач на различных языках (ПК-3);

- оптимизировать параллельные программы с учетом сложности задачи и специфики КС (ПК-6);

- организовывать выполнения параллельных программ на различных компьютерных платформах (ПК-3);

владеть:

- владеть информацией о современных компьютерах и компьютерных системах, их технических характеристиках и программном обеспечении (ПК-9);

- знаниями об основных проблемах развития компьютерных систем (ОПК-1);

- владеть актуальными публикациями по основным проблемам параллелизма и его реализации (ПК-6).

КРАТКОЕ СОДЕРЖАНИЕ РАЗДЕЛОВ ДИСЦИПЛИНЫ

Архитектура компьютеров и компьютерных систем

(14 часов самостоятельной работы)

Организация компьютера и методы повышения быстродействия процессоров и памяти. Сильно связанные (многопроцессорные КС, кластеры) и слабо связанные (локальные и глобальные) КС. Масштабирование, структуры связей и пропускная способность межузловых взаимодействий каналов. Предельные характеристики, проблемы достижения эффективного применения.

Параллелизм, формы его задания, критерии и параметры сложности, модели, распараллеливание последовательных программ

(18 часов самостоятельной работы)

Задачная и процессная формы задания параллелизма. Информационная независимость – фундаментальное условие распараллеливания на задачном уровне. Модель процессного параллелизма как множества порождаемых и взаимодействующих процессов. Модели параллельных процессов Р. Милнера, Ч. Хоара, сетей Петри, др. Критерии и параметры сложности параллелизма. Основные подходы к созданию языков для описания параллелизма и разработки параллельных программ.

Распараллеливание линейных участков последовательных программ и циклов. Использование условий для упреждающего распараллеливания. Средства описания параллелизма в последовательных программах: векторные команды, примитивы `fork` и `join`, примитивы OpenMP.

Функциональное параллельное программирование

(14 часов самостоятельной работы)

Парадигмы и функциональные языки, основанные на лямбда исчислении и операциях композиции функций: LISP, ML, Haskell (лямбда основанные), рекурсивные функции, FRTL (композиционные языки).

Задания параллелизма в лямбда основанных языках. Параллелизм в композиционных языках как следствие информационной независимости функций, вытекающее из свойств операций композиции функций. Язык композиционного функционального параллельного программирования FRTL. Средства разработки, анализа структурной и вычислительной сложности FRTL программ. Реализация на многопроцессорных (многоядерных) КС.

Нитевое программирование (10 часов самостоятельной работы)

Понятие нити и нитевой программы. Способы описания нитей с использованием языков последовательного программирования (C, C++, и др.). Взаимодействие нитей и синхронизация. Рекурсивное программирование нитей. Реализация нитевого параллельного программирования в MULTITHREADING. Сравнение с MPI, другими языками и средствами описания параллелизма.

Крупноблочное граф-схемное потоковое параллельное программирование и MPI (14 часов самостоятельной работы)

Компонентное параллельное программирование, основанное на явном задании информационных связей и только их между компонентами – основа языка граф-схемного параллельного программирования. Потоковая обработка и ее описание в граф-схемных параллельных программах. Интерпретация модулей граф-схемных программ с использованием последовательных языков. Параллельная операционная семантика языка граф-схемного потокового параллельного программирования. Реализация языка: среда программирования и средства управления процессами параллельного выполнения программ на КС.

Процессы и их взаимодействие в MPI. Описание параллельных процессов с использованием C++, др. языков. Средства отладки и организации выполнения на кластерах и распределенных КС. Средства описания порождаемых процессов в MPI-2. Сравнение MPI с другими языками и средами параллельного программирования.

Управление процессами выполнения параллельных программ на КС

(14 часов самостоятельной работы)

Задачи и функции управления параллельными процессами в КС. Методы и алгоритмы планирования параллельных процессов в КС. Методы и алгоритмы распределения ресурсов. Организационные схемы реализации управления КС: централизованное, децентрализованное и иерархическое управление.

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ КОНТРОЛЯ ОСВОЕНИЯ РЕЗУЛЬТАТОВ ОБРАЗОВАНИЯ ПО ДИСЦИПЛИНЕ

Промежуточная аттестация по итогам освоения дисциплины:

7 семестр – дифференцированный зачет.

Вопросы для самоконтроля

1. Способы классификации КС.
2. Критерий оценки предельных и реальных показателей эффективности КС.
3. Формы параллелизма в задачах и необходимые средства для их описания.
4. Проблемная ориентация различных средств задания параллелизма.
5. Что такое процесс с программной точки зрения?
6. Особенности моделей процессов Милнера, Хоара и сетей Петри.
7. Как выбирается оптимальная степень распараллеливания задач?
8. Как влияет степень распараллеливания на ускорение?
9. Почему рекурсия более сильное средство распараллеливания, чем итерация?
10. Основные приёмы и методы распараллеливания последовательных программ.
11. Связь ускорения, степени распараллеливания и сложности задач.
12. Ограничения MPI как средства распараллеливания.
13. В чём отличие нитей и ядер? Как выбирается количество нитей?

14. Основные механизмы управления процессами.
15. Критические факторы, влияющие на эффективность выполнения параллельных программ.
16. Какими параметрами определяется загруженность компьютеров и компьютерных программ?
17. Характеристики КС ТОП-500.
18. Свойства адаптивности и его применение в управлении процессами.
19. Как измеряется загруженность компьютеров и КС (память, каналов, процессоров)?
20. Как формулируется задача оптимального управления процессами на КС?

Вопросы для проведения зачета

1. Архитектура компьютера, его основные характеристики, способы увеличения быстродействия и производительности.
2. Режимы управления программами и процессами, реализуемая средствами ОС компьютера.
3. Архитектура КС, их технические характеристики, классификация.
4. Формы и характеристики параллелизма, способы представления и оценивания сложности.
5. Методы распараллеливания последовательных программ.
6. Модели параллельных процессов Мильнера, Хоара, сетей Петри.
7. Функциональные языки параллельного программирования: HASKEL, FRTL и др.
8. Аналитические модели сложности параллельного выполнения FRTL программ. Эквивалентное преобразование FRTL программ и сложность.
9. Средства параллельного программирования MULTITHREADING, сравнение с FRTL и MPI.
10. Средства параллельного программирования MPI для кластеров. Сравнение с языком граф-схемного программирования.
11. Язык граф-схемного параллельного программирования, сравнение с MPI и другими языками.

12. Модель параллельного выполнения граф-схемных потоковых параллельных программ и её реализация на КС.
13. Основные механизмы управления параллельными процессами на КС. Методы и алгоритмы планирования процессов.
14. Методы и алгоритмы управления загруженностью КС и оптимизация использования ресурсов.
15. Методы и алгоритмы адаптивного управления процессами и загруженностью в КС.
16. Методы, приёмы и технологии разработки параллельных программ.
17. Обеспечение отказоустойчивости и реконфигурирования КС.

Критерии оценки за освоение дисциплины определены в Инструктивном письме И-23 от 14 мая 2012 г.

РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

Основная литература:

1. В.П. Гергель. Современные языки и технологии параллельного программирования: учебник для вузов. Б-ка Нижегородского гос. ун-та им. Н.И. Лобачевского. – М.: Изд-во Моск. ун-та, 2012. – 408 с.
2. Левин М.П. Параллельное программирование с использованием OpenMP: учебное пособие. – М.: Интернет-Ун-т информ. технологий: БИНОМ. Лаборатория знаний, 2012. – 118 с.
3. И.И. Ладыгин, А.В. Логинов, А.В. Филатов, С.Г. Яньков. Кластеры на многоядерных процессорах: учебное пособие. – М.: Изд. дом МЭИ, 2008. – 112 с.

Дополнительная литература:

4. В.В. Воеводин, Вл.В. Воеводин. Параллельные вычисления: Научное издание: Учебное пособие для вузов. – СПб.: БХВ-Петербург, 2002. – 608 с.

5. Камерон Хьюз, Трейси Хьюз. Параллельное и распределенное программирование с использованием C++: пер. с англ. – М.: Вильямс, 2004. – 672 с.

6. Суперкомпьютерные технологии в науке, образовании и промышленности / В.А. Садовничий и др., Моск. гос. ун-т им. М.В. Ломоносова (МГУ), Суперкомпьютерный консорциум ун-тов России. РАН; Ред. В.А. Садовничий, и др. – М.: Изд-во Моск. ун-та, 2012. – 232 с.