

14. УЧЕБНЫЙ КУРС

СОВРЕМЕННЫЕ ЯЗЫКИ И ТЕХНОЛОГИИ ПАРАЛЛЕЛЬНОГО ПРОГРАММИРОВАНИЯ

ННГУ_ВМК_ФИИТ_П301 (ЯТ)

Цели и задачи курса

Учебный курс ориентирован на изучение широкого спектра алгоритмических языков и технологий, активно используемых при разработке параллельного программного обеспечения. В рамках курса рассматриваются технологии для разработки параллельных программ для систем с общей и распределенной памятью (OpenMP, MPI), языки параллельного программирования на основе разделенного глобально-адресуемого пространства (UPC, CFF, Chapel, X10) и технологии разработки параллельных программ для графических процессоров (CUDA, OpenCL). Освещение такого значительного набора языков и технологий в одном учебном курсе позволяет получить полное представление о разнообразии существующих подходов в области параллельного программирования.

Учебный курс может использоваться при обучении, как целиком, так и отдельными модулями в различных образовательных программах.

Требования к слушателям

Знание учебного материала курсов К101 «Введение в методы программирования-1», К102 «Введение в методы программирования-2», К201 «Методы программирования-2», К206 «Архитектура вычислительных систем», К202 «Операционные системы», П101 «Теория и практика параллельного программирования».

Описание курса

Основными учебными разделами курса являются:

ВВЕДЕНИЕ В ПАРАЛЛЕЛЬНОЕ ПРОГРАММИРОВАНИЕ

- **Общая характеристика параллельных вычислительных систем** (*Архитектура ЭВМ*)¹ – 1 час
Классификация вычислительных систем. Многопроцессорные системы с общей и распределенной памятью (мультипроцессоры и мультикомпьютеры).
Схемы коммутации. Анализ параллельных алгоритмов и типовые топологии схем коммутации.
- **Основные понятия параллельного программирования** (*Операционные системы*)¹ – 1 час
Понятия процессов, потоков, ресурсов. Параллельная программа как система процессов и потоков. Взаимодействие, взаимоисключение, синхронизация и взаимоблокировка.
Передача данных для организации взаимодействия процессов. Основные операции передачи данных.
Показатели эффективности параллельных вычислений.

БАЗОВЫЕ ТЕХНОЛОГИИ ПАРАЛЛЕЛЬНОГО ПРОГРАММИРОВАНИЯ (Теория и практика параллельных вычислений)¹

- **Технология разработки параллельных программ для многопроцессорных систем с общей памятью (стандарт OpenMP)** – 1 час
Общая характеристика стандарта OpenMP. Создание параллельных областей. Разделение вычислительной нагрузки между потоками. Работа с данными. Синхронизация. Функции и переменные окружения. Сравнительная характеристика подходов параллельного программирования для систем с распределенной и общей памятью.

¹ Данный раздел предназначен для повторения (закрепления) ранее пройденного учебного материала – в скобках приводится названия курса, в котором был первоначально рассмотрен повторяемый материал.

- **Технология разработки параллельных программ для многопроцессорных систем с распределенной памятью (стандарт передачи сообщений MPI) – 1 час**

Общая характеристика стандарта MPI. Режимы передачи данных. Коллективные операции. Конструирование производных типов данных. Управление процессами. Создание логических топологий. Примеры: матричные вычисления, решение уравнений в частных производных.

ТЕХНОЛОГИИ РАЗРАБОТКИ НА ОСНОВЕ СУЩЕСТВУЮЩИХ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ

- **Язык параллельного программирования UPC – 2 часа**

Модель параллельного программирования Unified Parallel C (UPC). Понятие UPC-программы. Модель представления памяти. Управление распределением данными. Синхронизация вычислений в UPC. Коллективные операции. Оценка эффективности. Примеры.

- **Язык параллельного программирования CAF – 2 часа**

Модель параллельного программирования Co-Array Fortran (CAF). Понятие исполнителей (images). Понятие распределенных массивов (co-arrays). Синхронизация параллельных вычислений. Оценка эффективности CAF-программ. Развитие: от CAF 1.0 к CAF 2.0. Примеры.

НОВЫЕ ЯЗЫКИ ПАРАЛЛЕЛЬНОГО ПРОГРАММИРОВАНИЯ НА ОСНОВЕ РАЗДЕЛЕННОГО ГЛОБАЛЬНО-АДРЕСУЕМОГО ПРОСТРАНСТВА

- **Язык параллельного программирования Chapel – 3 часа**

Модель разделенного глобально-адресуемого пространства – язык Chapel. Элементы языка. Параллелизм задач. Расширенные возможности работы с данными. Понятие исполнителя (locale). Управление распределением данных. Примеры.

- **Язык параллельного программирования X10 – 3 часа**

Основные понятия X10. Понятие исполнителей (places) и активностей (activities). Поддержка параллелизма в X10. Управление распределением данных. Примеры.

ТЕХНОЛОГИИ ПАРАЛЛЕЛЬНОГО ПРОГРАММИРОВАНИЯ ДЛЯ ГРАФИЧЕСКИХ ПРОЦЕССОРОВ

- **Технология CUDA – 1 час**

Модель программирования. Модель исполнения и иерархия потоков. Иерархия памяти. Спецификаторы типов переменных и функций. Встроенные переменные.

Конфигурирование исполнения ядер. Синхронизация. Управление устройствами. Управление памятью.

Общие принципы вычислений на базе технологии CUDA. Примеры.

- **Технология разработки на основе стандарта OpenCL – 1 час**

Архитектура OpenCL. Модели платформы, среды исполнения, памяти и программирования. Общие принципы вычислений на базе OpenCL.

Перенос приложения CUDA на OpenCL. Примеры.

ДОПОЛНИТЕЛЬНЫЕ ПОДХОДЫ К РАЗРАБОТКЕ ПАРАЛЛЕЛЬНЫХ ПРОГРАММ

- **Алгоритмический язык параллельного программирования ОККАМ – 1 час**

Концепция процесса в ОККАМ. Элементарные и составные процессы. Методы конструирования агрегированных процессов (последовательное и параллельное исполнение, условие, выбор и селекция процессов, цикл и репликация). Принципы передачи данных между процессами при помощи каналов. Примеры.

- **Алгоритмический язык Фортран HPF для параллельного программирования – 1 час**

Базовые типы данных. Конструирование производных типов данных. Массивы. Сечения. Динамические массивы. Конструкторы. Директивы распределения данных.

Структура программы. Понятие подпрограммы. Модули и интерфейсы. Управление параллельными вычислениями. Примеры.

- **Технология разработки параллельных программ DVM – 1 час**

Распределение данных. Локализация и удаленные данные. Многомерные массивы.

Распараллеливание циклов. Процедуры. Методика отладки DVM-программ. Примеры.

- **Язык параллельного функционального программирования T++ – 1 час**

Язык T++ — расширение C++. T-переменные, T-функции, T-указатели. Структура программы на языке T++. Общая память (суперпамять). Передача данных.

Готовые и неготовые величины. Ожидание готовности. Гранулы параллелизма. Использование глобальных переменных. Примеры.

Учебно-методическое обеспечение курса содержит:

- комплект электронных учебных материалов;
- презентации для проведения лекционных занятий;
- комплект лабораторных работ.

Ожидаемые результаты обучения

Обучаемые, успешно освоившие учебный курс, будут:

- знать современные подходы к разработке языков и технологий параллельных программ;
- владеть рассмотренными в учебном курсе языками и технологиями параллельного программирования, понимать их преимущества и недостатки;
- уметь разрабатывать параллельные программы для вычислительных систем различной архитектуры (системы с общей и распределенной памятью, системы с разделенным глобально-адресуемым пространством памяти, системы с графическими процессорами общего назначения);
- иметь навыки самостоятельного изучения новых разрабатываемых языков и технологий параллельного программирования.

Учебная литература

1. Гергель В.П. Теория и практика параллельных вычислений. М.: Интернет-Университет Информационных технологий; Бинوم, 2007. 424 с.
2. Гергель В.П. Современные языки и технологии параллельного программирования. М.: Изд-во Московского университета, 2012. 408 с.
3. Линев А.В., Боголепов Д.К., Бахраков С.И. Технологии параллельного программирования для процессоров новых архитектур: Учебник / Под ред. В.П. Гергеля. М.: Изд-во Московского университета, 2010. 160 с.

Сайт учебного курса: <http://www.hpcc.unn.ru/?dir=804>

Контактная информация: декан ф-та ВМК, д.т.н., проф. В.П. Гергель, gergel@unn.ru