

Аннотации к лекциям

В. Карпов "Основы распараллеливания программ"

В настоящее время развитие вычислительных систем испытывает третий кризис программного обеспечения. Первый кризис разразился в 60–70е годы прошлого века, когда программирование в машинных кодах и на языке ассемблера вошло в противоречие с возросшей производительностью компьютеров. Выходом стало появление языков высокого уровня. Второй кризис пришелся на 80–90е годы. Создание и поддержка сложных и надежных программных комплексов, содержащих несколько миллионов строк кода, написанных сотнями программистов, потребовали развития объектно-ориентированных языков и разработки инструментария для поддержки больших программных проектов. Третий кризис связан с невозможностью дальнейшего экстенсивного развития hardware и переходом к многоядерным архитектурам. Адекватного ответа на возникший кризис до сих пор не найдено. Одним из способов его преодоления является разработка параллельных программ.

Для разработки параллельных программ существует два принципиально разных подхода. Первый из них заключается в создании новых алгоритмов, которые будучи не эффективными на стандартных последовательных архитектурах позволяют получить при решении возникающих задач значительной эффективности на архитектурах параллельных. Второй подход — это попытка адаптации хорошо исследованных последовательных алгоритмов для новых архитектур. Именно второму подходу посвящен данный курс.

Лекция 1

туальность параллельной обработки данных

Определение суперкомпьютера, история параллельности, классификация вычислительных систем, асимптотический анализ, декомпозиция.

Лекция 2

Асимптотический анализ

Декомпозиция, асимптотический анализ, строгая параллельная и каноническая форма графа.

Лекция 3

ктивности и виды зависимостей

Активности и атомарные операции, наборы активностей, условие Бернштейна, зависимости во вложенных циклах, зависимости в невложенных циклах.