

Аннотация рабочей программы дисциплины  
по направлению 01.03.02 Прикладная математика и информатика  
профиль: Математическое моделирование

## Технология высокопроизводительных вычислений

### Цели и задачи освоения дисциплины

Основной целью изучения дисциплины является углубить и систематизировать знания студентов в области современных технологий параллельного программирования OpenMP, MPI, CUDA, позволяющие разрабатывать приложения эффективно работающие на современных вычислительных платформах, в том числе, с гибридной архитектурой. Особое внимание уделено использованию специализированных математических библиотек, адаптированных для вычислений на различных архитектурах и разработке параллельных приложений для решения прикладных задач.

Основные задачи изучения дисциплины заключаются в приобретении знаний, умений и владений, благодаря которым студенты, используя современные технологии высокопроизводительных вычислений, могут эффективно применять вычислительные платформы с гибридной архитектурой для решения практических задач.

### Место дисциплины в структуре ОПОП ВО

Дисциплина «Технология высокопроизводительных вычислений» относится к обязательной части образовательной программы по направлению 01.03.02 Прикладная математика и информатика.

Дисциплина преподается в 8 семестре, на 4 курсе. Форма промежуточной аттестации – зачет.

Перечень дисциплин с указанием разделов (тем), предшествующих дисциплине «Технология высокопроизводительных вычислений»:

Семестр	Дисциплина	Разделы
2	Архитектура вычислительных систем	Практическое введение в технологию MPI
		Практическое введение в технологию OpenMP
4	Численные методы	Численные методы решения задач линейной алгебры
		Методы решения нелинейных уравнений и систем нелинейных уравнений

## Содержание дисциплины

№	Содержание раздела
Раздел 1	<p><b>Высокопроизводительные платформы и тенденции их развития, технологии высокопроизводительных вычислений.</b></p> <p>Базовые понятия, используемые в технологиях высокопроизводительных вычислений и смежных технологиях.</p> <p>Краткая история развития высокопроизводительных вычислений.</p> <p>Вводное занятие по работе на гетерогенной вычислительной платформе ЛИТ ОИЯИ <i>HybriLIT</i> : структура платформы, порядок работы в режиме удаленного доступа, доступное программное обеспечение, использование <i>Modules</i>, планировщика задач <i>SLURM</i> (написание script-файлов) для запуска приложений в режиме очереди.</p>
Раздел 2	<p><i>Технология параллельного программирования OpenMP.</i></p> <p>Компиляторы Intel с поддержкой OpenMP. Инструментальные средства разработки и отладки многопоточных приложений.</p> <p><i>Архитектура Intel Xeon Phi</i> второго поколения, технологии параллельных вычислений, векторизация.</p> <p>Библиотека <b>Intel® Math Kernel Library</b>.</p> <p>Задачи на использования функций Intel MKL: векторные операции, умножение матриц</p> <p>Библиотека Intel MKL: ScaLAPACK – решение систем линейных алгебраических уравнений.</p> <p>Компиляторы Intel с поддержкой OpenMP. Инструментальные средства разработки и отладки многопоточных приложений.</p>
Раздел 3	<p>Технология параллельного программирования <i>Message Passing Interface (MPI)</i>: разработка параллельных программ для проведения расчетов на системах с распределенной памятью.</p> <p>MPI-реализации программ умножения плотно заполненных квадратных матриц (распределение на одномерную сетку процессов, распределение на двумерную решетку процессов).</p> <p>Организация коллективной рассылки данных, Организация глобальной операции с помощью MPI_Reduce.</p> <p>Программа приближенного вычисления определенного интеграла (вычисление числа <math>\pi</math>).</p> <p>Разработка MPI-приложения вычисления двойного интеграла с заданной точностью. Проведение анализа эффективности параллельной программы.</p>
Раздел 4	Гибридные технологии: MPI+OpenMP
Раздел 5	<p><b>Высокопроизводительные вычисления с применением графических процессоров (GPU). Технология Nvidia CUDA.</b></p> <p>Введение в вычисления общего назначения с использованием GPU (парадигма GPGPU). Основные архитектурные отличия GPU от CPU.</p> <p>Архитектура современных GPU (на примере Nvidia Tesla K80). Введение в технологию CUDA: модель памяти CUDA, типы памяти, понятия потока.</p> <p>CUDA-расширение языка C (спецификаторы функций, спецификаторы переменных, встроенные переменные, директивы запуска ядра).</p> <p>Параллельные математические библиотеки: <i>CUBLAS</i>, <i>CUFFT</i>, <i>CUSPARSE</i>.</p> <p><i>Технология Nvidia CUDA</i>. Шаблон работы с разделяемой памятью.</p> <p>Оптимизация работы с разделяемой памятью.</p> <p>CUDA-реализация программы нахождения суммы модулей комплексного вектора с использованием разделяемой памяти.</p>

	CUDA-реализация программы умножение матриц с использованием функций библиотеки CUBLAS.
Раздел 6	<b>Технология Nvidia CUDA.</b> Использование pinned-памяти, CUDA-потoki ( <i>CUDA-streams</i> ). Использование в расчетах нескольких GPU. OpenMP+CUDA приложения. Вычислительные схемы и их параллельные реализации на гибридных вычислительных архитектурах задач математической физики.