

Аннотация рабочей программы дисциплины

Б1.О.32 «Параллельное и низкоуровневое программирование»

Направление

подготовки/специальность 02.03.02 Фундаментальная информатика и информационные технологии

(код и наименование направления подготовки/специальности)

Курс 3 Семестр 5 Количество з.е. 2

Объем трудоемкости: 2 зачетных единиц (72 ч., из них – 50 час. аудиторной нагрузки: лекционных 16 ч., лабораторных работ - 34 ч., 19,8 часов самостоятельной работы, 2 часов КСР, 0,2 часа ИКР.), форма контроля – зачет.

Цель дисциплины: формирование у студентов знаний и практических навыков разработки многопоточных и высокопроизводительных приложений на языке C++ с использованием технологий POSIX Threads, OpenMP, CUDA, ROCm и OpenCL. Освоение архитектурных принципов параллельных вычислений на CPU и GPU, а также низкоуровневой оптимизации под современные процессоры (SSE, AVX).

Задачи дисциплины:

1. Изучить основные принципы параллельного программирования и синхронизации потоков.
2. Освоить API POSIX Threads и средства параллелизма C++.
3. Научиться использовать OpenMP для распараллеливания вычислений на уровне потоков и циклов.
4. Изучить принципы векторизации вычислений с применением SSE и AVX-инструкций.
5. Освоить основы GPU-программирования на CUDA, ROCm и OpenCL.
6. Развить навыки профилирования и оптимизации кода под архитектуру CPU и GPU.

Место дисциплины в структуре ООП ВО:

Дисциплина «Параллельное и низкоуровневое программирование» относится к базовой части блока Б1 **Дисциплины (модули)**.

Дисциплина в качестве пререквизитов требует уверенного владения языком C/C++ (указатели, управление памятью, рекурсия, функции), знаний базовых алгоритмов и структур данных (оценка сложности, массивы, списки, деревья), а также понимания принципов работы операционных систем (процессы, потоки, синхронизация, планировщик). Постреквизитами выступают дисциплины, опирающиеся на эффективные параллельные и низкоуровневые вычисления, среди которых «Технологии обработки больших данных» (распределённая и многопоточная обработка), «Современные методы компьютерного зрения» (GPU-свёртки, параллельная фильтрация), «Подготовка данных машинного обучения» (векторизация и ускорение на CPU/GPU), а также «ИИ в робототехнике» (вычисления реального времени, low-level оптимизация).

Связи разделов дисциплины с пререквизитами и постреквизитами:

Связь	Дисциплина (актуальное название из учебного плана)	Раздел(ы) текущей РПД, зависящие от данной дисциплины	Характер зависимости
Пререквизит	Программирование	Разделы 2–5 (работа с потоками, указатели, память в C/C++), разделы 6–7 (написание GPU-ядер)	Критический (без C/C++ невозможно)

Пререквизит	Операционные системы	Раздел 1 (потoki vs процессы), раздел 2 (синхронизация POSIX), раздел 3 (mutex, deadlock), раздел 8 (профилирование)	Сильный
Постреквизит	Технологии обработки больших данных	Раздел 1 (параллельная обработка), разделы 2–4 (многопоточное ускорение), раздел 8 (масштабирование)	Сильный
Постреквизит	Технологии компьютерного зрения	Раздел 5 (SIMD для фильтров), раздел 6 (CUDA для свёрток), раздел 7 (OpenCL/ROCm), раздел 8 (оптимизация под GPU)	Сильный
Постреквизит	Современные технологии машинного обучения	Раздел 5 (векторизация матричных операций), раздел 6 (GPU-ускорение нейросетей), раздел 8 (профилирование пайплайнов)	Сильный

Результаты обучения (знания, умения, опыт, компетенции):

Содержание и структура дисциплины:

ОПК-2	Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности
ОПК-2.1	Знает основные положения и концепции в области программирования, архитектуру языков программирования, теории коммуникации, знает основную терминологию, знаком с содержанием Единого Реестра Российских программ Знает концепции параллельного программирования, демонстрирует знание архитектуры языков (C/C++), использует корректную терминологию (поток, mutex, deadlock и т.д.), показывает понимание стандартов и реестров программ (POSIX, OpenMP, CUDA/ROCm).
ОПК-3	Способен к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям
ОПК-3.1	Аргументировано применяет методы проектирования, разработки и реализации программных продуктов и программных комплексов в различных областях человеческой деятельности Знает и аргументированно объясняет решения по структуре программы и способам распараллеливания.
ОПК-3.2	Использует инструментальные, программные и аппаратные средства измерений для оценки качества программного обеспечения Умеет выбирать подходящие методы проектирования параллельных приложений, реализует лабораторные работы с pthreads, std::thread, OpenMP, CUDA

ОПК-6	Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности
ОПК-6.1	Аргументировано применяет современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения Умеет использовать отечественные и международные инструменты (компиляторы, библиотеки, профайлеры), обосновывает выбор технологий для лабораторных работ и проектов.
ОПК-6.2	Ориентируется в современных положениях и концепциях прикладного и системного программного обеспечения, архитектуры компьютеров и сетей (в том числе и глобальных), технологии создания и сопровождения программных продуктов и программных комплексов Владеет способами применения современных технологии параллельного программирования и GPU/CPU оптимизации
PL-4	Способен применять языки программирования C/C++ для решения задач в области ИИ
PL-4.1	Разрабатывает и отлаживает эффективные многопоточные решения на C++, тестирует, испытывает и оценивает качество таких решений Знает основы синтаксиса языка. Знает общие принципы параллельных вычислений и понимает проблемы, возникающие при распараллеливании алгоритмов. Проводит распараллеливание простого алгоритма с применением OpenMP, стандартных библиотек C/C++ или др. встроенных систем.
PL-4.2	Разрабатывает и отлаживает системы ИИ на C++ под конкретные аппаратные платформы с ограничениями по вычислительной мощности, в том числе для встроенных систем Решает проблемы одновременного доступа к данным из нескольких потоков, грамотно применяет атомарные операции и механизм блокировок. Оценивает производительность, умеет профилировать код и устраняет найденные узкие места. Понимает возможности и ограничения
PL-4.3	Разрабатывает и отлаживает решения на C++, использующие GPU и FPGA для массовой параллелизации вычислений в рамках общей системы ИИ, с применением как готовых решений, так и разработкой своих Находит и использует библиотеки, соответствующие решаемой задаче. Знает основы синтаксиса языка C/C++, основы построения систем ИИ, общие принципы параллельных вычислений. Умеет использовать средства отладки и профилирования кода, находить участки кода, ограничивающие производительность системы.

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.

Разделы дисциплины, изучаемые в 5 семестре (очная форма)

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	
1	2	3	4	5	6	7
1.	Введение в параллельное программирование. Потоки и процессы в ОС.	8	2		4	2

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	СРС
1	2	3	4	5	6	7
2.	POSIX Threads (pthreads): создание, управление и синхронизация потоков.	8	2		4	2
3.	Параллелизм в C++: std::thread, mutex, future, async.	8	2		4	2
4.	OpenMP: директивы распараллеливания, секции, редукции, профилирование	8	2		4	2
5.	Векторизация и оптимизация вычислений: SIMD, SSE, AVX.	8	2		4	2
6.	GPU-программирование: основы архитектуры CUDA, модель потоков и памяти.	8	2		4	2
7.	OpenCL и ROCm: кроссплатформенные вычисления на GPU.	8	2		4	2
8.	Оптимизация и профилирование параллельных приложений на CPU и GPU.	13,8	2		6	5,8
ИТОГО по разделам дисциплины		69,8	16		34	19,8
Контроль самостоятельной работы (КСР)		2				
Промежуточная аттестация (ИКР)		0,2				
Подготовка к текущему контролю		0				
Общая трудоемкость по дисциплине		72				

Примечание: Л – лекции, КСР – контрольные и самостоятельные работы, ЛР – лабораторные занятия, СРС – самостоятельная работа студента

Курсовые проекты или работы.

Не предусмотрены учебным планом

Вид аттестации: ЛР, контрольная работа, зачет.

Основная литература

1. Богачёв К.Ю. Основы параллельного программирования: учебное пособие. – М.:Лаборатория знаний, 2020. – 345 с.
2. Лаптев В.В. C++ Объектно-ориентированное программирование: Учебное пособие. – СПб.: Питер, 2008 – 464 с.
3. Алексеев, В.Е. Структуры данных. Модели вычислений / В.Е. Алексеев, В.А. Таланов. - 2-е изд., исправ. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 248 с. : схем., ил. - (Основы информационных технологий)
4. Эндрюс Г.Р. Основы многопоточного, параллельного программирования. М.:Издательский дом “Вильямс”. 2003. 512с
5. Малявко А.А. Параллельное программирование на основе технологий OpenMP, CUDA, OpenCL, MPI: учебное пособие для вузов / А. А. Малявко. - 3-е изд., испр. и доп. - Москва : Юрайт, 2022. - 135 с. - URL: <https://urait.ru/bcode/492127> (дата обращения: 29.11.2022). - Режим доступа: для авториз. пользователей. - ISBN 978-5-534-14116-0. - Текст: электронный.

Автор Приходько Т.А. – кандидат технических наук, доцент кафедры вычислительных технологий;

Нигодин Е.А. – старший преподаватель кафедры вычислительных технологий.