

# Параллельное программирование

Лекторы:



**Язык:**

Русский

**Трудоемкость:**

3 з.е.

**Форма контроля:**

Диф. зачет

**Образовательная программа:**

[Беспроводные технологии](#)

8 семестр

**Прerequisites:**

[Программирование](#)

Лекции (ак.час)*	Практические занятия (ак.час)	Лабораторные занятия (ак.час)
32	32	
*1 академический час = 45 минутам		

Цель курса -- познакомить студентов с проведением параллельных вычислений. В курсе студенты освоят основы проведения численных расчётов на графических процессорах, освоят программирование кода для графических процессоров NVIDIA с использованием CuPy, Numba и PyCUDA в среде языка программирования Python, освоят принципы и паттерны построения параллельных алгоритмов, программирования для кода многопроцессорных машин с использованием OpenMP. По окончании курса студенты смогут применить полученные навыки для ускорения программного кода на Python с использованием графических процессоров и программного кода на C/C++ с использованием технологии OpenMP.

## Содержание курса

### План курса

#### Структура курса

Раздел	Содержание
Основы параллельных вычислений на графических процессорах	<ul style="list-style-type: none"><li>- Введение в курс.</li><li>- Архитектура и устройство графических процессоров.</li><li>- Знакомство с программно-аппаратной платформой CUDA.</li></ul>
Программирование кода для графических процессоров	<ul style="list-style-type: none"><li>- Основы программирования с использованием JIT-компилятора Numba.</li><li>- Программирование кода для графических процессоров на Python с использованием Numba.</li><li>- Создание универсальных функций <code>ufunc</code> для GPU с использованием <code>cuda.vectorize</code>.</li><li>- Создание обобщенных универсальных функций для GPU с использованием <code>cuda.guvectorize</code>.</li><li>- Написание и запуск ядер для GPU общего вида.</li><li>- Оптимизация и работа с памятью графических процессоров.</li><li>- Обзор возможностей <code>cyru</code> и <code>PyCUDA</code>.</li></ul>
Программные библиотеки для вычислений с использованием GPU	<ul style="list-style-type: none"><li>- Библиотека <code>cuBLAS</code>: линейная алгебра на графическом процессоре.</li><li>- Библиотека <code>cuFFT</code>: преобразование Фурье на графическом процессоре.</li><li>- Библиотека <code>cuSPARSE</code>: ускорение вычислений с разреженными матрицами.</li><li>- Библиотека <code>cuRAND</code>: генерация случайных чисел.</li></ul>
Программирование кода для многопроцессорных машин	<ul style="list-style-type: none"><li>- Алгоритмы CUDA Sorting.</li><li>- Введение в архитектуру многопроцессорных машин.</li><li>- Таксономия Флинна: SISD, SIMD, MISD, MIMD.</li><li>- Параллельные вычисления на многопроцессорных машинах с использованием OpenMP.</li></ul>

#### Рекомендуемые ресурсы

- 1) Fundamentals of Accelerated Computing with CUDA Python (онлайн-курс). <https://courses.nvidia.com/courses/course-v1:DLI+C-AC-02+V1/about>.
- 2) Программирование GPU при помощи Python и CUDA, Б. Тоуманен / пер. с англ. А. В. Борескова. – М.: ДМК Пресс, 2020. – 254 с.
- 3) Параллельные вычисления общего назначения на графических процессорах : учебное пособие / К. А. Некрасов, С. И. Поташников, А. С. Боярченко, А. Я. Купряжкин. — Екатеринбург : Изд-во Урал. ун-та, 2016. — 104 с.
- 4) Основы программирования на CUDA : учебно-методическое пособие / В. В. Коробицын, Ю. В. Фролова. – Омск : Изд-во Ом. гос. ун-та, 2016. – 68 с.

#### Политика оценивания

1. Домашнее задание - 20 баллов - дедлайн 3 неделя
2. Лабораторная работа - 30 баллов - дедлайн 6 неделя
3. Дифференцированный зачёт - 50 баллов - дедлайн 12 неделя