



Казанский
федеральный
университет

ВЫСШАЯ ШКОЛА
информационных технологий
и информационных систем

Библиотеки OpenCL

Boost.Compute

- ▶ Библиотека предоставляющая C++ интерфейс для работы с многоядерными CPU и GPU
- ▶ Основана на OpenCL
- ▶ <https://github.com/boostorg/compute>
- ▶ При компиляции кода необходимо указать
 - -I/путь к папке с boost/
- ▶ Требуемая версия Boost ≥ 1.54
- ▶ Включен в Boost начиная с версии 1.61



Boost.Compute

- ▶ Пример: получение дефолтного устройства OpenCL



Boost.Compute

- ▶ Идеология применения библиотеки схожа с STL – операции над контейнерами данных
- ▶ В библиотеке реализовано несколько видов контейнеров для работы с данными на устройстве
- ▶ Список алгоритмов:
 - http://www.boost.org/doc/libs/1_62_0/libs/compute/doc/html/boost_compute/reference.html#boost_compute.reference.api_overview.algorithms



Boost.Compute

- ▶ Пример: передача данных с хоста на устройство посредством Boost.Compute
- ▶ Обработка данных на устройстве через функцию `transform()`
- ▶ Передача обработанных данных с устройства на хост



Boost.Compute

- ▶ Библиотека предоставляет возможность создавать пользовательские функции, для передачи в алгоритмы типа `transform()` и `reduce()`
- ▶ Boost.Compute поддерживает лямбда-выражения для передачи в соответствующие алгоритмы в качестве параметра



Boost.Compute

- ▶ Пример: использования собственных функций и лямбда-выражений для передачи в `transform()`



VexCL

- ▶ Шаблонная библиотека векторных выражений
- ▶ Предназначена для облегчения разработки кода с использованием GPU
- ▶ В качестве бэкенда используются OpenCL/CUDA
- ▶ Поддержка вычислений на нескольких устройствах и платформах



VexCL

- ▶ Библиотека предоставляет STL-подобный интерфейс для работы с контейнерами и алгоритмами с использованием параллельного программирования
- ▶ <https://github.com/ddemidov/vexcl>
- ▶ Для компиляции проекта необходимо
 - Указать папку с библиотеками Boost(флаг -L)
 - Слинковать его с boost_system и boost_filesystem



VexCL

- ▶ Пример: получение списка GPU поддерживающих вычисления в двойной точностью и включение их в текущий контекст



VexCL

- ▶ Основной фокус библиотеки – операции над векторами
 - Векторы должны иметь одинаковый размер
 - Должны размещаться в памяти одного и того же устройства
- ▶ Каждая операция над векторами приводит к запуску вычислительного ядра



VexCL

- ▶ Пример сложения векторов
- ▶ Пример создания пользовательской функции
- ▶ Пример сортировки



Задание на практику

- Редукция с использованием библиотеки VexCL