

Поддержка кортежей и структур в библиотеке Brahma.FSharp

Автор: Лунина П. С., 243 группа

Руководитель: к. ф. - м. н. Григорьев С. В.

Санкт-Петербургский Государственный Университет

18.05.2017

Введение

- GPGPU — техника использования GPU для общих вычислений, обычно выполняемых CPU
- OpenCL — одна из реализаций этой техники
- Библиотека Brahma.FSharp, основанная на транслировании F# quotation в OpenCL, позволяет использовать технику GPGPU в программах, написанных на F#
- В настоящий момент в библиотеке Brahma.FSharp не реализована поддержка кортежей и структур

Цель

Реализация поддержки кортежей и структур в библиотеке Brahma.FSharp.

Задачи

- Изучение библиотеки Brahma.FSharp
- Изучение особенностей языка OpenCL
- Реализация поддержки структур и кортежей в библиотеке Brahma.FSharp:
 - Трансляция объявления типов-структур с FSharp на OpenCL
 - Трансляция создания новых структур и обращения к полям по имени с FSharp на OpenCL
 - Создание типов-структур для кортежей на стороне OpenCL
 - Трансляция функций работы с кортежами с FSharp на OpenCL и создания нового кортежа
- Тестирование
- Добавление примеров и описания на официальный сайт

Brahma.FSharp

- Brahma.FSharp — библиотека на F# для интеграции вычислений на GPGPU
- В Brahma.FSharp реализованы трансляция, механизм работы с памятью, а также некоторые функции для обработки примитивных типов данных и массивов
- Частично реализована поддержка структур, однако она практически не протестирована и работает не всегда правильно
- Было бы полезно наличие поддержки кортежей и структур для параллельных вычислений на GPGPU

OpenCL

- OpenCL — фреймворк для написания программ, связанных с параллельными вычислениями на различных центральных и графических процессорах
- В OpenCL можно создавать, объявлять и инициализировать структуры
- Полем структуры не может быть массив
- Кортежи в OpenCL отсутствуют, однако кортеж можно задать как структуру

Реализация поддержки структур

- Объявление структуры на F# транслируется в соответствующее объявление на OpenCL
- При инициализации нового экземпляра структуры на стороне OpenCL создается объект специально созданного типа StructType с доступом к полям структуры, который переводится в структуру, поддерживаемую OpenCL
- Структура загружается в буфер памяти соответствующего размера

Реализация поддержки структур

- Проведенные тесты показали, что существующая в библиотеке реализация трансляции объявления типов-структур и обращения к полям работает правильно
- Исправлены ошибки при инициализации новой структуры
- Реализованы трансляция создания новых структур внутри kernel функции и создание структур с несколькими конструкторами от разного количества аргументов
- Реализована возможность использования массивов структур

Реализация поддержки кортежей

- Кортеж каждого типа задан как структура с именем tuple + уникальный номер
- В трансляторе создан тип TupleType, представляющий собой надстройку над типом StructType с зафиксированными именами полей
- При загрузке в память динамически определяются типы данных элементов кортежа и создается экземпляр специальной структуры с полями обобщенного типа, конструктор которой может принимать любые переменные

Реализация поддержки кортежей

- В данный момент в библиотеке доступны кортежи из двух и трех элементов, однако по аналогии можно расширить возможности и для большего количества
- Реализованы передача кортежей в качестве аргументов и создание новых кортежей внутри функции
- Реализована возможность использования массивов кортежей
- Для кортежей из двух элементов реализована возможность доступа к элементам с помощью стандартных функций `fst` и `snd`
- Для кортежей из трех элементов - с помощью отдельно реализованных функций `first`, `second` и `third`

Результаты

- Изучены возможности библиотеки Brahma.FSharp
- Изучены особенностей языка OpenCL
- В библиотеке Brahma.FSharp реализованы:
 - Трансляция объявления типов-структур с FSharp на OpenCL
 - Трансляция создания новых структур и обращения к полям по имени с FSharp на OpenCL
 - Создание типов-структур для кортежей (на стороне OpenCL)
 - Трансляция функций работы с кортежами с FSharp на OpenCL и создания нового кортежа
- Проведено тестирование в системе NUnit
- На официальный сайт добавлены примеры и описание