

Третье совместное рабочее совещание  
ИПС имени А. К. Айламазяна РАН  
и  
МГТУ имени Н. Э. Баумана  
по функциональному языку программирования Рефал  
(г. Москва, 12 июля 2020 г.)

Программа

Время	Название доклада	Докладчик
	12 июня 2020 г.	
12:00	Открытие	
12:05	Первая сессия (председатель: Антонина Непейвода)	
12:05–12:35	<b>Древесные оптимизации в компиляторе Рефала-5λ: итоги и перспективы</b> ( <a href="#">аннотация</a> )	Коновалов Александр
12:40–13:10	<b>Использование отношения Хигмана-Крускала для прерывания рекурсивной специализации функций</b> ( <a href="#">аннотация</a> )	Андрей Кошелев
13:15–13:45	<b>Автоматическая разметка оптимизируемых функций</b> ( <a href="#">аннотация</a> )	Елена Калинина
13:45	перерыв	
14:00	Вторая сессия (председатель: Александр Коновалов)	
14:00–14:30	<b>Языки образцов в анализе Рефал-программ</b> ( <a href="#">аннотация</a> )	Антонина Непейвода
14:35–15:05	<b>Распознавание экранируемых предложений в программе на Рефале-5λ</b> ( <a href="#">аннотация</a> )	Александр Барлука
15:10–15:40	<b>Оптимизация вызовов встроенных функций в компиляторе Рефала-5λ</b> ( <a href="#">аннотация</a> )	Ханага Агазаде
15:40	Обсуждение	
16:00	Закрытие	

## Аннотации к докладам

Александр Коновалов

«Древесные оптимизации в компиляторе Рефала-5λ: итоги и перспективы»

Компилятор Рефала-5λ поддерживает две оптимизации, работающие на уровне синтаксического дерева: специализация и прогонка. Оба механизма оптимизируют вызовы помеченных функций.

Оптимизация специализации строит для каждого вызова специализируемой функции новый экземпляр функции, в теле которого учитывается информация о вызове, известная на момент компиляции. Оптимизация прогонки пытается встроить вызываемую функцию в вызывающую.

Компилятор позволяет выводить в файл журнала синтаксическое дерево программы до и после оптимизации в виде псевдокода на Рефале, благодаря чему древесные оптимизации позволяют использовать Рефал-5λ для простых экспериментов по трансформации программ.

В докладе рассказывается о том, на что способны эти оптимизации, а также про перспективы их дальнейшего развития.

Андрей Кошелев

«Использование отношения Хигмана-Крускала для прерывания рекурсивной специализации функций»

В ряде случаев применение специализации функций может привести к заикливанию компилятора.

В работе рассматривается решение проблемы заикливания рекурсивной специализации. Для этого предлагается использовать отношение Хигмана-Крускала, применяемое при решении похожей проблемы в суперкомпиляции.

Елена Калинина

«Автоматическая разметка оптимизируемых функций»

Высокоуровневые языки программирования, в том числе и Рефал-5λ, содержат в себе много абстрактных программистских конструкций — они делают процесс написания кода для человека максимально удобным, но снижают скорость работы программы. Поэтому компиляторы стараются автоматически оптимизировать код. Есть множество разных видов оптимизаций, но мы сосредоточимся на двух из них: прогонке и специализации. Они уже реализованы в компиляторе Рефала-5λ, однако маркировка функций, которые нужно оптимизировать по-прежнему остаётся задачей программиста. Автоматическая разметка сняла бы эту ответственность и позволила бы больше фокусироваться на написании поддерживаемого кода, чем заниматься ручными оптимизациями. По сути программист сможет сосредоточиться на решении конкретных задач, оставив монотонный однотипный труд компьютеру.

Темой данного доклада и будет являться добавление в компилятор Рефала-5λ корректной и безопасной автоматической разметки прогоняемых и специализированных функций.

## Аннотации к докладам (продолжение)

Антонина Непейвода

«Языки образцов в анализе Рефал-программ»

Языки образцов (pattern languages) исследуются в computer science с 1980-х годов, в первую очередь в контексте поиска наиболее точного образца, язык которого включает заданное множество строк. Доклад представляет собой обзор некоторых из этих исследований — они оказываются применимы для анализа рефал-программ. В частности, будет рассмотрено, какие свойства левых частей рефал-программы могут указывать на хорошую применимость к ней стандартных методов суперкомпиляции, а какие — на возможность плодотворного использования альтернативного подхода, использующего современные наработки в области языков образцов.

Александр Барлука

«Распознавание экранируемых предложений в программе на Рефале-5λ»

Предложение в программе может не выполняться, если его левая часть (образец+условия) описывает множество аргументов, являющихся подмножеством другого предложения, записанного выше. В таких случаях мы будем говорить, что одно предложение экранирует другое. Экранирующие предложения такого вида в исходных текстах программ могут быть написаны только по ошибке.

Предлагается добавить в компилятор Рефала-5λ проверку предупреждений об экранировании для упрощения отладки и написания программ.

Ханага Агазде

«Оптимизация вызовов встроенных функций в компиляторе Рефала-5λ»

Встроенными функциями (intrinsic functions) называются функции, которые распознаются компилятором и транслируются особым образом, как правило ради повышения производительности.

В докладе рассматриваются встроенные библиотечные функции Рефала-5(λ), вызовы которых можно вычислять на этапе компиляции и особенности реализации этого механизма.