

ОГЛАВЛЕНИЕ

ОГЛАВЛЕНИЕ.....	4
ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ.....	5
ВВЕДЕНИЕ	6
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	9

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

ПО – программное обеспечение.

ЯП – язык программирования.

ВВЕДЕНИЕ

В области разработки программного обеспечения довольно часто поднимается вопрос производительности разрабатываемых программ. Требование к производительности — одно из важнейших нефункциональных требований для большинства продуктов.

На пути от написания кода программы до исполнения соответствующего ей машинного кода на процессоре компьютера пользователя есть множество факторов, которые так или иначе могут влиять на производительность разрабатываемой программы.

Не углубляясь в конкретные факторы, выделим стадии с этими факторами от написания кода до исполнения машинного кода программы на процессоре:

- написание кода программистом,
- компиляция кода (некоторые стадии могут отсутствовать, либо быть совмещены):
 1. лексический анализ,
 2. синтаксический анализ,
 3. семантический анализ,
 4. оптимизации на абстрактном синтаксическом дереве,
 5. трансляция в промежуточное представление,
 6. машинно-независимые оптимизации,
 7. трансляция в конечное представление (машинный код),
 8. машинно-зависимые оптимизации.
- интерпретация или компиляция «на лету» промежуточного кода виртуальной машиной (при трансляции кода компилятором не в машинный код, а в код некоторой виртуальной машины),
- планирование исполнения машинного кода ядром операционной системы,

- исполнение кода программы на процессоре,
- исполнение кода функций библиотек поддержки времени исполнения, используемых в программе,
- исполнение кода системных вызовов, используемых в программе.

На каждой из перечисленных стадий существует множество факторов, способных в конечном счете повлиять на производительность программы. Под контролем программиста непосредственно целевой программы находится лишь одна группа факторов. За остальные группы факторов ответственны разработчики соответствующих инструментов и вспомогательных программ или их частей: виртуальных машин, библиотек поддержки времени исполнения, ядер операционных систем и непосредственно самих процессоров и других физических компонентов, способных влиять на производительность исполняемой программы.

В данной работе предлагается провести исследование, связанное с анализом влияния на производительность программ групп факторов в рамках стадий написания кода и его компиляции.

Хочется сразу отметить, что результаты такого анализа могут быть использованы как разработчиками целевых программ, так и разработчиками языка программирования, на котором эти программы составляются.

В качестве языка программирования, программы на котором и компилятор которого будут исследоваться, был выбран Kotlin, как один из наиболее интересных и быстро развивающихся языков.

Таким образом, целью данной магистерской диссертации является разработка набора инструментов для анализа исходного кода программ на ЯП Kotlin и выявления потенциальных проблем производительности в них, а также проведение исследования по данной теме с использованием разработанного набора инструментов.

Практическая значимость работы заключается в получении по результатам исследования и разработки:

1. сгруппированного списка файлов с исходным кодом из достаточно объемного набора данных, являющихся с точки зрения тех или иных алгоритмов аномальным;
2. набора инструментов, позволяющего получать аналогичный список файлов на заданном проекте.

Первый результат должен стать важным и полезным в первую очередь разработчикам языка программирования Kotlin; второй же — пользователям языка — программистам, использующих для разработки своих проектов язык программирования Kotlin.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ...

