

Министерство образования и науки Российской Федерации
Московский физико-технический институт
(национальный исследовательский университет)

Физтех-школа радиотехники и компьютерных технологий
Кафедра микропроцессорных технологий в телекоммуникационных сетях и
вычислительных системах

Выпускная квалификационная работа бакалавра по направлению 03.03.01
«Прикладные математика и физика»

Исследование и разработка системы анализа кода
для повышения производительности программ на
языке программирования высокого уровня

Студентка Б01-008 группы
Лирисман К. С.

Научный руководитель
Гаврин Е. А.

Долгопрудный
2024

Содержание

Аннотация	v
1. Введение	1
2. Постановка задачи	3
3. Обзор существующих решений	5
4. Теоретическая часть	7
5. Практическая часть	9
6. Заключение	11
Литература	13

Аннотация

В настоящий момент активно развивается статически типизированный управляемый язык программирования, являющийся расширенной и более быстрой версией языка TypeScript (далее TS). Основная идея разработки спецификации и компилятора этого языка программирования — сделать его максимально похожим на TS для упрощения перехода будущих разработчиков между TS и выбранным для исследования языком, а также для ускорения переписывания существующих на TS приложений. Таким образом, между целевым языком и TS формируется общая часть, — корректная с точки зрения TS. Оставшуюся часть называют не поддерживаемой TS.

В данной работе предлагается выделить из целевого ЯП подмножество, наиболее выгодное с точки зрения производительности, в том числе за счёт отсекающего удобного для разработчиков, но медленного функционала. Таким образом, пока сам язык неминуемо развивается в сторону общности с TS'ом ради легкости перехода, предложенная система помогает держать фокус на производительности.

Анализ происходит в момент компиляции исходного кода и предлагает включение желаемых проверок группами или по отдельности путем добавления флагов компиляции. Реализовано 8 пунктов проверки, опирающихся на спецификацию выбранного языка программирования высокого уровня и реализацию его компилятора. К общим с TS проверкам относятся: неявная упаковка и распаковка, ускорение проверок равенства, запрет инструкций верхнего уровня, исключая классы и функции, предложение установки модификатора для класса или метода как финального и другие. Предлагается использование двух режимов работы проверяющей системы — в состоянии предупреждений, а именно, предложений, не обязывающих разработчика к исправлению замечаний и не влияющих на результат работы программы, и в состоянии, приводящему к ошибке при ненулевом количестве предложений, ожидающих от пользователя последующих исправлений, и считающегося за ошибку компиляции. Предложенная система имеет некоторые варианты для повышения производительности. Однако существуют сценарии использования, когда разработчикам необходимо отключение этих проверок. Разработчики могут активировать или деактивировать любую опцию по одной или включать группами. Также можно снять с проверки определенные строки или целые части кода с помощью предложенного аналога системы точечного отключения проверок Clang Tidy непосредственно в исходном коде программы в виде многострочных или однострочных комментариев. Таким образом,

данная система значительно повышает скорость работы и время запуска приложения на выбранном статически типизированном языке высокого уровня, позволяет разработчикам простыми методами улучшить их код и потенциально избежать некоторых ошибок.

Глава 1

Введение

Здесь идет текст. Вот так выглядит ссылка на библиографию. Аналогично добавляются еще главы, внутри них можно объявлять секции с помощью `\section`.

Глава 2

Постановка задачи

Еще текст.

Глава 3

Обзор существующих решений

И СНОВА ТЕКСТ.

Глава 4

Теоретическая часть

так тоже неплохо.

Глава 5

Практическая часть

практикуем.

Глава 6

Заключение

закключаем

Литература

Будет добавлена.