

# Отладка макросов в языке программирования Scala

Найданов Дмитрий, 545 группа

Научный руководитель: А.Н. Терехов

Технический консультант: Е.Н. Бурмако

Рецензент: Д.Ю. Булычев

# Язык Scala

- Мультипарадигменный язык программирования для JVM
- Разработан в LAMP EPFL
- Версия 2.10 вышла 4 января 2013
- Среди новой функциональности есть макросы

# Макросы

- Макрос — некоторый программный объект, который во время вычисления заменяется на новый объект
- Определение макроса в Scala — это фактически подключаемый модуль компилятора
- Генерация кода «на лету», DSL, реификация, различные оптимизации
- Есть во многих языках программирования: Lisp, OCaml, Nemerle

# Отладка макросов в языке Scala

- Код, управляющий логикой раскрытия, можно отлаживать обычным способом
- С кодом, который получается после раскрытия, есть проблемы:
  - Где взять (синтетический) исходный код для абстрактных синтаксических деревьев (AST)?
  - Как исправить отладочную информацию?

# Постановка задачи

Разработать и реализовать подсистему компилятора и модуль некоторого отладчика для отладки кода, полученного после раскрытия макроса

# Обзор подхода

- На стороне компилятора:
  - Сгенерировать синтетический код
  - Сгенерировать байт-код с правильным отображением
  - Сохранить синтетический код для отладчика
- На стороне отладчика:
  - Загрузить синтетический код
  - Использовать загруженный код для обработки точек прерывания, трассировки (stepping) и т. п.

# Сторона компилятора

- Подсистема реализована в виде отдельной фазы компилятора
- После фазы проверки типов происходит генерация синтетического кода для раскрытых макросов
- Сразу же создается файл для исходного кода и переписываются позиции у AST
- Файл и упакованные позиции помеченных деревьев передаются в стандартный поток вывода

# Переписывание позиций AST

- После создания файла с синтетическим кодом нужно изменить позиции AST, чтобы они указывали на новый файл
- Два основных подхода
  - Использования механизма установки позиций парсера на синтетическом коде
  - Обход AST и переписывание позиций «вручную»
- Был выбран второй подход



# Сторона отладчика

- Выбран встроенный отладчик IntelliJ IDEA
- Принимает синтетический код, сгенерированный компилятором
- Создает и использует виртуальный файл

# Результаты

- Разработан подход для отладки кода, полученного от раскрытия макроса в компиляторе языка Scala
- Реализованы механизмы для отладки на стороне компилятора и на стороне встроенного отладчика в IntelliJ IDEA