Insert here your thesis' task.

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Master's thesis

Fine-tuning LLVM transformation passes

Bc. Tomáš Drbota

Department of Theoretical Computer Science
Supervisor: doc. Ing. Ivan Šimeček, Ph.D.

June 1, 2018

Acknowledgements

THANKS (remove entirely in case you do not with to thank anyone)

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended. In accordance with Article 46(6) of the Act, I hereby grant a nonexclusive authorization (license) to utilize this thesis, including any and all computer programs incorporated therein or attached thereto and all corresponding documentation (hereinafter collectively referred to as the "Work"), to any and all persons that wish to utilize the Work. Such persons are entitled to use the Work in any way (including for-profit purposes) that does not detract from its value. This authorization is not limited in terms of time, location and quantity.

In Prague on June 1, 2018 .....

Czech Technical University in Prague
Faculty of Information Technology

Citation of this thesis

Drbota, Tomáš. *Fine-tuning LLVM transformation passes.* Master's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2018.

Abstrakt

V několika větách shrňte obsah a přínos této práce v českém jazyce.

Klíčová slova Replace with comma-separated list of keywords in Czech.

Abstract

Modern compilers provide a wide range of different optimizations in an effort to increase the performance of the resulting program. Many of them also provide a way to customize the level of optimization (as an example -O flags in the GCC toolchain, or in-line attributes for loop unrolling), but the granularity of these approaches is low.

The purpose of this work is to explore and analyze options for customizing the execution of transformation passes in the LLVM compiler system. The intent is to create ways for the user to be able to specifically declare which transformations should or should not be used in a given scope, e.g. function. The majority of the implementation will be done on the closely related LLVM C/C++ frontend, Clang. This fine-tuned code will then be compared to automatically optimized code, and the results analyzed.

Keywords llvm, clang, compiler, transformation, optimization

Contents

List of Figures

Introduction

Text will go here.

Chapter 1

State-of-the-art

Chapter 2

Analysis and design

Chapter 3

Realisation

Conclusion

Bibliography

Appendix A

Acronyms

todo TODO

Appendix B

Contents of enclosed CD

- `readme.txt` ..... the file with CD contents description
- `exe` ..... the directory with executables
- `src` ..... the directory of source codes
  - `wbdcm` ..... implementation sources
  - `thesis` ..... the directory of LaTeX source codes of the thesis
- `text` ..... the thesis text directory
  - `thesis.pdf` ..... the thesis text in PDF format
  - `thesis.ps` ..... the thesis text in PS format