TECHNISCHE
UNIVERSITÄT
DARMSTADT

DEEDS
Group

Master's Thesis

DeterministiC: Taming Undefined Behavior of C Compilers

Background

A significant fraction of virtually every computer's software stack is written in the C programming language, which is specified by an ISO standard. This specification of the language, however, is incomplete. As a consequence, it is possible to write C code that is valid according to the standard but whose behavior is not unambiguously defined.

Such *undefined behavior* is problematic, because the definition of the resulting program behavior is left to the compiler implementers and, even worse, they are free to alter this behavior any time. Therefore, any program that contains statements with undefined behavior can break if it gets compiled with a different compiler or a new compiler version than the one(s) it had been tested with. Unfortunately, such undefined behavior is by no means limited to exotic corner cases, but affects simple operations like pointer comparisons or integer arithmetic (the behavior in case of integer overflows is undefined). Thus, a large fraction of C code is probably affected by the aforementioned problem.

Objectives

The goal of this thesis is twofold. In a first step, the problem of undefined behavior needs to be systematically assessed. The C standard needs to be scanned for undefined behavior and the usage of these constructs empirically assessed. Differences of the implemented behavior across different compiler versions need to be identified.

In a second step, a compiler plugin will be implemented to control the compiler's handling of undefined behavior. The goal is to enable C programmers to explicitly specify the behavior they expect from undefined behavior. With this plugin, compilers should produce identically behaving code, even if the implementation of undefined behavior changes over time, to allow for a safe reuse of legacy C code.

Prerequisites

Candidates should be able to understand and write C and C++ code. Experience with gcc or Clang plugin implementations are beneficial. The thesis will be written in English.

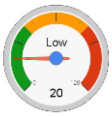
Duration/Start

Immediate

Contact

Dr. Stefan Winter
sw@cs.tu-darmstadt.de
+49 6151 16 25226
S2|02 E221

Literature



Analysis

Implementation

Awesomeness

Last updated: 2017-03-01