Improving Register Usage

Abstract

Petr Bednář (xbedna46@stud.fit.vutbr.cz)
Jan Ťulák (xtulak00@stud.fit.vutbr.cz)

November 8, 2015

Performance of operations on modern machines have been improving quicker than the performance of the attached memory. As a result, every access to memory means delays of operations and an unfavorable overall performance. Improving the register usage is an important factor for RISC architectures, where most operations work only with registers.

Problem of variables to register allocation is commonly solved by graph coloring, which tries to allocate as many variables to registers as possible. But this approach will fail when optimizing access to a memory location, which is not defined by a scalar variable, but an expression, which is a common case for array access in the C programming language.

In this talk, we discuss improving register usage by source to source compiler transformations. We discuss multiple algorithms of such transformations, which are useful, especially for repeated access to array items, common in loops.