C to WebAssembly Analysis

Still underway, but I promise

it will be very cool ฅ^•ﻌ•^ฅ

Sneak peak

Abstract

WebAssembly is an increasingly popular, portable, low-level byte-code language designed to work with JavaScript in the browser. WebAssembly allows browsers to support CPU-intensive applications, and the Emscripten compiler makes it possible to cross-compile traditional C/C++ desktop applications to the web. Currently, transparent porting is still

hard to accomplish - practitioners need to modify the application source code to make it fit the WebAssembly architecture. In this paper, we investigate what misbehavior can occur after porting C programs to WebAssembly. We collect 14,024 valid C programs from three sources, compile them into WebAssembly using Emscripten, and compare their execution outputs with those of the original C programs. 54 of these programs exhibit different execution outputs.

We manually inspected these discrepant results and identified nine root causes. For each type of cause, we provide advice to developers on how to handle them in practice. In conclusion, porting existing C programs to WebAssembly for cross-platform distribution still needs code adaptations. More effort should be put into stitching inconsistencies between WebAssembly and other high-level languages