BACHELOR THESIS: FROM TOUCHDEVELOP TO IMPERATIVE PROGRAMS

INTRODUCTION

TouchDevelop (http://research.microsoft.com/en-us/projects/touchdevelop/) is a novel programming language introduced by Microsoft to develop scripts on mobile devices (e.g., smartphones). Therefore, the TouchDevelop programming language is particularly simple, since it has to allow one to develop full programs using a touchscreen, and it has to target hardware with limited resources. For instance, there is no dynamic allocation of memory, and the core of the language is a simple imperative language. On the other hand, TouchDevelop programs interact with all the components of a mobile device (e.g., the GPS sensor and the songs stored in the memory card) through some libraries. www.touchdevelop.com contains a huge collection of existing TouchDevelop scripts.
Since the TouchDevelop IDE does not provide any feedback to the user in case of runtime errors, static analysis could help the developers to debug their application. In particular, we could run the static analysis in the cloud, and then give to the developer some feedback about possible bugs.
During the last three years, at the Chair of Programming Methodology we have developed the static analyzer Sample, which is based on the abstract interpretation theory. This analyzer has already been applied to a wide range of properties (for instance, types, string values, integers, heap structures) and to different programming languages (Scala and Java bytecode). Sample works on an intermediate language (called Simple) aimed at expressing the most important object-oriented features with a minimal set of statements. In order to apply Sample to TouchDevelop scripts, we need to translate these scripts to the language analyzed by Sample.

GOAL

The goal of this project is to develop the translation of the TouchDevelop language to Simple. While the main object-oriented features are not required by TouchDevelop, the system libraries adopted to model the mobile system will require to deeply study how these features can be represented in a standard imperative language. This will be the main focus of this thesis. The student is expected to implement the translation as well. Note that, since TouchDevelop already provides an interface to access the abstract syntax tree of a program, the student will not be required to write a parser of TouchDevelop scripts.

This thesis is in the context of the project “TouchBoost – Cloud-based Static Analyses to Improve Lay Programming on Mobile Devices” funded by Microsoft.