Project documentation - Kast Tree Interface

Gheorghita Mutu

December 2017

1 Introduction

KAST, abbreviating K abstract syntax tree or KAST, is a generic, abstract syntax notation for the user-defined concrete syntax in K sentences. When writing sentences using KAST, users have to replace concrete language syntax, e.g.

if (B) S1 else S2

with syntax in labelled form, e.g.

if(_)_else_(B,S1,S2)

KAST serves two major purposes: (1) it is an escape mechanism to avoid parsing ambiguities in sentences; and (2) it can be parsed using the generic K parser in one pass, avoiding the expensive and complex step of generating parsers for the user-defined syntax combined with the generic K syntax.

2 Project Purpose

This project has the purpose of making a graphical interface for K's abstract tree generated by KAST command.

3 Implementation Details

The code required for this program will be written in C and it is intended to be cross-platform. For the actual graphic interface it will be used either QT - for cross-platform - or if there won't be enough time it will be used IDA's wingraph32.exe* which just gets a specific input and builts a graphic tree. Under the hood there will be a linked-list implementation which will link token alike structures, unary and binary operators structures with pointers to their arguments, lists structures with pointers to all elements, etc.

4 Bibliography

https://github.com/kframework/k/wiki/KAST-and-KORE

5 Notes

* IDA states that "The freeware version of IDA Pro can not be used commercially. The full version is not limited." for the version 6 from which I'm getting the wingraph32.exe.