Project outside course scope

Jacob Herbst(mwr148), Matilde Broløs (jtw868)

Introduction Language Notes

An imperative language with static verification

Advisor: Ken Friis Larsen

Project outside course scope

Jacob Herbst(mwr148), Matilde Broløs (jtw868)

Introduction Language Notes

An imperative language with static verification

Advisor: Ken Friis Larsen

*

Abstract

Contents

IFC is a small imperative programming language with a build in assertion language enabeling the possibility for statically verifying programs by the use of external provers. In this section we will present the syntax and semantics of the imperative language and the assertion language.

⟨*statement*⟩ ::= ⟨*ident*⟩ '`=`' ⟨*expr*⟩
| '`for`' ⟨*ident*⟩ '`=`' ⟨*expr*⟩ '`to`' ⟨*expr*⟩ '`do`' ⟨*statement*⟩
| '`{`' ⟨*stat-list*⟩ '`}`'
| ⟨*empty*⟩

⟨*stat-list*⟩ ::= ⟨*statement*⟩ '`;`' ⟨*stat-list*⟩ | ⟨*statement*⟩

Abstract: Udpensl delen om forventningen til projektet. Hvad er den korte version om alt hvad vi har lavet (forklar det til en hypotetisk Fritz) Background - husk weakest precondition vs wlp vs strongest postcondition Designvalg - fx method 1 vs method 2 Spekulationer om udvidelse til PER - er det nemt at udvidde det vi har skrevet Concurrency? Et nondeterministisk valg mellem s1 og s2? At tænke over hvor nemt det er at udvidde er en måde at lave en evaluering ar vores arbejde
Skal vi skrive noget contribution? Så skal vi have nogle overvejelser at basere det på