

Sammanfattning

Denna D-uppsats syftar till att beskriva bakgrunden till och tillkomsten av en utvecklingsmiljö för Java som stöder kontraktsprogrammering. Uppsatsen specificerar vad syntax och semantik är. Denna kunskap krävs för att till fullo förstå uppsatsens mål och syfte. Begreppet kontraktsprogrammering förklaras ingående då detta är ett nyckelbegrepp i utvecklingsmiljön och därmed denna uppsats. En genomgång av ett antal verktyg som används inom området kontraktsprogrammering görs och testas för att demonstrera deras funktionalitet. Uppsatsen beskriver sedan ett idealsystem som vore det bästa möjliga resultatet av arbetet. Detta idealsystem används sedan för att jämföra det egentliga resultatet av arbetet med. Resultatet är en utvecklingsmiljö för Java som stöder kontraktsprogrammering genom att presentera för- och eftervillkor vid metदानrop. Utvecklingsmiljön stöder även presentation av klassinvarianter samt automatiskt infogande av förvillkorstest. Denna utvecklingsmiljö ska förenkla för programmeraren att använda sig av kontraktsprogrammering vid utveckling av mjukvara. Författarnas förhoppning är att denna utvecklingsmiljö ska kunna bidra till bättre och mer pålitlig mjukvara.

Development Environment for Java with Support for Programming by Contract

Abstract

The purpose of this master's thesis is to describe the background to and creation of a development environment for Java that supports programming by contract. The thesis specifies what syntax and semantics are since this is needed to understand the rest of the document. The concept of programming by contract is thoroughly explained since this is a key notion in the development environment and therewith also in this thesis. A review is made of some of the tools that are used within the area of programming by contract. These tools are also tested to demonstrate their functionality. The thesis then describes an ideal system that would be the best possible result of the project. This ideal system is later on used to compare the actual result of the project with. The result is a development environment for Java that supports programming by contract by showing pre- and postconditions when invoking a method. The development environment also supports showing class invariants and automatic insertion of tests on preconditions. The intent of this development environment is to simplify for the programmer to use contracts in software development. It is the authors' hopes that this will lead to better and more reliable software.