Universidade do Minho
Escola de Engenharia

Abstract Interpretation

Ferramentas e Técnicas

Paulo Gameiro **A72067**
Pedro Rodrigues **PG41092**
Rafaela Soares **A79034**

Universidade do Minho
Escola de Engenharia

Introdução

Em grande parte dos casos, a validação de software parece ser a fase mais dispendiosa e complexa de todo o processo de desenvolvimento.

Existem diversas métricas de software que possibilitam realizar o planeamento destes projectos. A partir destas métricas, pode-se identificar a quantidade de esforço, de custo, etc.

Universidade do Minho
Escola de Engenharia

Abstract Interpretation

O conceito de Abstract Interpretation também é conhecido como Deep Flow analysis, é um método formal matemático baseado em técnicas formais de verificação que proporcionam aos programas abstrair a semântica do software.

A verificação de programas consiste na melhoria da semântica do programa que satisfaz a sua especificação.

Um exemplo de abstracção de semântica é a lógica de Hoare, enquanto que os exemplos de abstracção de especificação são o invariante, correcção parcial e total.

Universidade do Minho
Escola de Engenharia

Abstract Interpretation

A abstração deve ser sólida - a prova de que a semântica abstrata satisfaz a especificação abstrata implica que a semântica concreta também satisfaz a especificação concreta.

Uma abstracção fraca pode levar a falsos negativos, ou seja, o programa pode ser dado como correcto em relação à especificação, quando este é, de facto, incorrecto.

A abstracção deve ser também completa e não podem ser deixados de fora aspectos da semântica.

Universidade do Minho
Escola de Engenharia

Abstract Interpretation

Em todos os casos, a abstracção de semântica deve ser sólida, completa e simples, de forma a evitar explosões combinatórias.

Os métodos formais são interpretações abstratas com maneiras diferentes de abstrair as semânticas. Em todos os casos, as semânticas abstratas têm de ser escolhidas para serem representadas por computadores.

Universidade do Minho
Escola de Engenharia

Abstract Interpretation

Métodos:

- Em model-checking, a abstracção de semântica é fornecida manualmente pelo utilizador na forma de um modelo finito de execuções do programa.
- Nos métodos dedutivos (deductive methods) a abstracção semântica é especificada por condições de verificação e têm de ser fornecidas pelo utilizador na forma de propriedade que se mantêm verdadeiras à medida que o programa é executado
- Em análise estática, a abstracção de semântica é calculada automaticamente graças a aproximações predefinidas, também possivelmente parametrizadas pelo utilizador

Universidade do Minho
Escola de Engenharia

Abstract Interpretation

Para se interiorizar melhor esta métrica, foque-se na equação -3 * 4 * 5 = ?

Não é necessário que esta seja resolvida para se ter conhecimento de que o resultado é um número negativo, se se recorrer às regras do sinal de multiplicação.

Universidade do Minho
Escola de Engenharia

Abstract Interpretation

Desta forma, é possível detectar hipotéticos problemas de confiabilidade no software, uma vez que existem ferramentas de interpretação abstrata que permitem detectar automaticamente diversos tipos de erros de programação

Exemplos de possíveis erros que podem ser identificados são overflows de buffers, conexões de bases de dados não fechadas, não referência a null pointer, entre outros não menos relevantes.

Universidade do Minho
Escola de Engenharia

Ferramentas

Julia Analyzer

JULIASOFT
code analysis reinvented

Pré-requisitos de utilização:

- Conhecimento em Java, Android ou .NET(C#).
- Experiência num dos seguintes ambientes de desenvolvimento integrado: Eclipse, Visual Studio, IntelliJ ou Android Studio.
- Criar uma conta em https://portal.juliasoft.com/.
- Seguir a documentação do IDE escolhido, disponibilizado em "Quickstart" no menu do utilizador em https://portal.juliasoft.com/, no intuito de instalar a ferramenta com sucesso.

Universidade do Minho
Escola de Engenharia

Ferramentas

Clang Static Analyzer

Pré-requisitos de utilização:

- Conhecimento em C ou C++.
- Instalação do Clang, através de http://clang-analyzer.llvm.org/ ou através do comando “sudo apt-get install clang”, no Linux.

Universidade do Minho
Escola de Engenharia

Ferramentas

Polyspace Code Prover

Pré-requisitos de utilização:

- Conhecimento em C ou C++.
- Solicitar o teste gratuito em https://www.mathworks.com/products/polyspace-code-prover.html.

Universidade do Minho
Escola de Engenharia

Ferramentas

SPARTA

- C++
- API simples e que podem ser facilmente montados para construir um analisador estático
- Permite obter os resultados da análise de uma forma mais eficiente e escalável

Universidade do Minho
Escola de Engenharia

Ferramentas

Coverity

- C, C++, Java, Javascript, entre outras...
- Gratuita para software open-source
- Bom suporte em diferentes compiladores
- Fácil configuração de análises estáticas

Universidade do Minho
Escola de Engenharia

Ferramentas

CodeSonar

- Permite analisar fluxos de dados e examinar o processamento de todo o programa que está a ser analisado

- Possibilidade de integração no Julia Analyser
  - Maior precisão
  - Análises mais avançadas de múltiplas linguagens numa única interface
  - Facilidade de colaboração entre equipas.

Universidade do Minho
Escola de Engenharia

Ferramentas

C++Test

- C++
- É uma simples e poderosa Framework de testes unitários para realizar testes automáticos
- Focada na usabilidade

Universidade do Minho
Escola de Engenharia

Considerações finais

As métricas de software possibilitam avaliar e identificar quantitativamente o esforço e custos das atividades que vão ser realizadas.

Um exemplo destas é a interpretação abstracta. Esta métrica de qualidade de software permite reduzir os custos e tem como objectivo fornecer uma teoria básica de forma a que os diversos métodos e ferramentas possam compreender tudo numa única framework.

As ferramentas variam desde programas com interface a bibliotecas que analisam o código submetido. No entanto, a utilização de algumas ferramentas não foi possível, como foi o caso da Polyspace Code Prover, Coverity, CodeSonar e C++Test/BugDetective, pelo facto de não se ter conseguido obter uma versão gratuita.

Universidade do Minho
Escola de Engenharia

Abstract Interpretation

Ferramentas e Técnicas

Paulo Gameiro **A72067**
Pedro Rodrigues **PG41092**
Rafaela Soares **A79034**