MFES - Métodos Formais em Engenharia de Software

Ficha da Disciplina

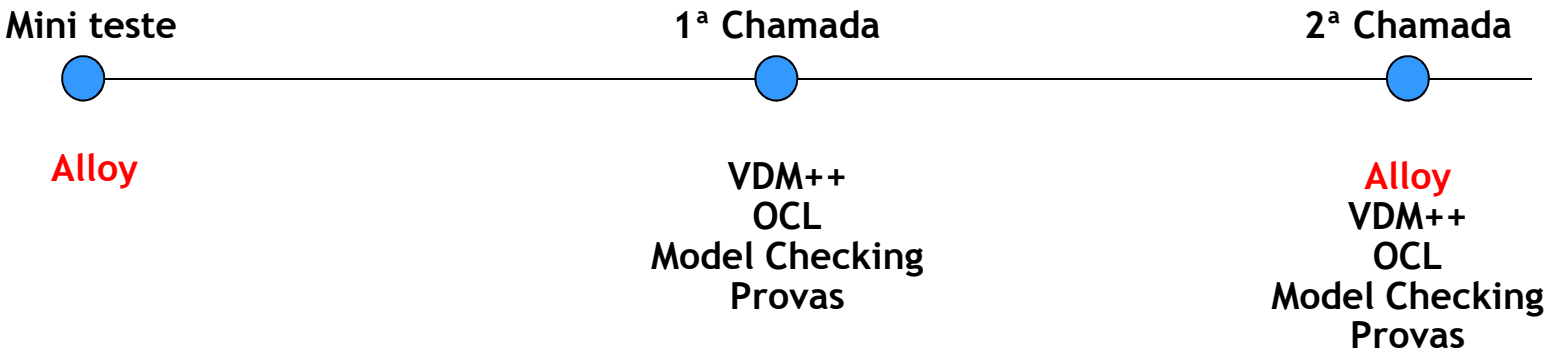
Ana Paiva

apaiva@fe.up.pt **www.fe.up.pt/~apaiva**

Ficha da disciplina - Avaliação

- Avaliação distribuída com exame final, com as seguintes componentes:
  - a) mini teste de Alloy, com consulta, duração de 1h, peso **40%**, nota mínima de 45%.
  - b) trabalho prático de VDM++, peso **25%**, nota mínima de 45%.
  - c) exame final com consulta, duração 1h30, peso **35%**, nota mínima de 45%.
- Nota:
  - A classificação final não pode exceder em mais de 3 valores a classificação do exame arredondada para o inteiro mais próximo.
  - Os alunos que não obtiverem aprovação no mini teste de Alloy poderão fazer um módulo de avaliação adicional no exame de recurso.
  - A melhoria da componente de Alloy só poderá ser melhorada no exame de recurso.

Ficha da disciplina - Avaliação



Ficha da disciplina - Avaliação

- Avaliação especial (TE, DA, …)
  - Os trabalhos são obrigatórios para TODOS os alunos, mesmo para os alunos dispensados de frequência às aulas
- Melhoria de Classificação Final/Distribuída
  - A classificação do exame pode ser melhorada em exame de recurso
  - A classificação obtida no trabalho prático pode ser melhorada na edição seguinte da disciplina
  - A classificação obtida no mini teste pode ser melhorada em exame de recurso.

Ficha da disciplina - Bibliografia

Software Abstractions, Daniel Jackson, http://mitpress.mit.edu/catalog/item/default.asp?ttype=2&tid=10928

Validated designs for object-oriented systems, Fitzgerald, John, Springer, 2004, ISBN: 1-85233-881-4

The Object Constraint Language, Second Edition, Jos Warmer e Anneke Kleppe; ISBN: 978-0-321-17936-4

- Proof and Disproof in Formal Logic, Richard Bornat, Oxforn University Press, 2005, ISBN: 0-19-8530269
- Systems and Software Verification - Model Checking Techniques and Tools, B. Bérard; M. Bidoit; A. Finkel; F. Laroussinie; A. Petit; L. Petrucci; Ph. Schnoebelen; P. McKenzie, Springer, 2001, ISBN: 3-540-41523-8

Ficha da disciplina - Bibliografia

- Model Checking, Clark, Jr., Edmund M.; Grumberg, Orna; Peled, Doron A., Cambridge, MA : MIT Press, 1999, ISBN: 0-262-03270-8
- Modelling Systems—Practical Tools and Techniques in Software Development, John Fitzgerald, Peter Gorm Larsen, Cambridge University Press, 1998, ISBN: 0-521-62605-6
- Specification of Software Systems, V.S. Alagar, K. Periyasamy, Springer, 1998, ISBN: 0387984305

Programa - Visão Geral

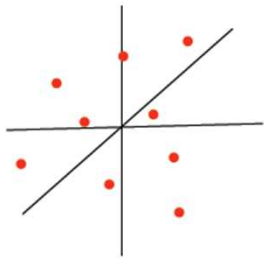
- *Model Finder* (Alloy)
- Especificação baseada em modelos, teste e geração de código (VDMTools)
- *Model checking*
- Provas formais (Coq, Jape Caduceus (C), Krakatoa (java))

Programa

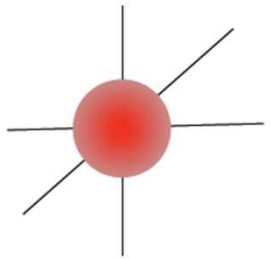
- 1. Introdução
  - O que são métodos formais.
  - Importância e aplicabilidade dos métodos formais no desenvolvimento de software.
  - Modelos de ciclo de vida e processos de desenvolvimento de software incorporando métodos formais.
  - Especificação, refinamento, implementação, verificação e validação.
  - Classificação de métodos formais.
  - Modelos explícitos vs implícitos, executáveis vs não executáveis.
  - Técnicas de verificação formal.

Programa

- 2. "Alloy Constraint Analyzer", para modelação e análise semântica
  - Modelação declarativa.
  - Diferenças relativas a "model checking".
  - Comandos Alloy.
  - Funções; Predicados; Factos; Asserções e Verificações ("Checks").
  - Modelação estática vs dinâmica.
  - Simular a execução de uma operação.
  - Verificar propriedades "safety".
- A ferramenta Alloy analyzer.



testing:
a few cases of arbitrary size

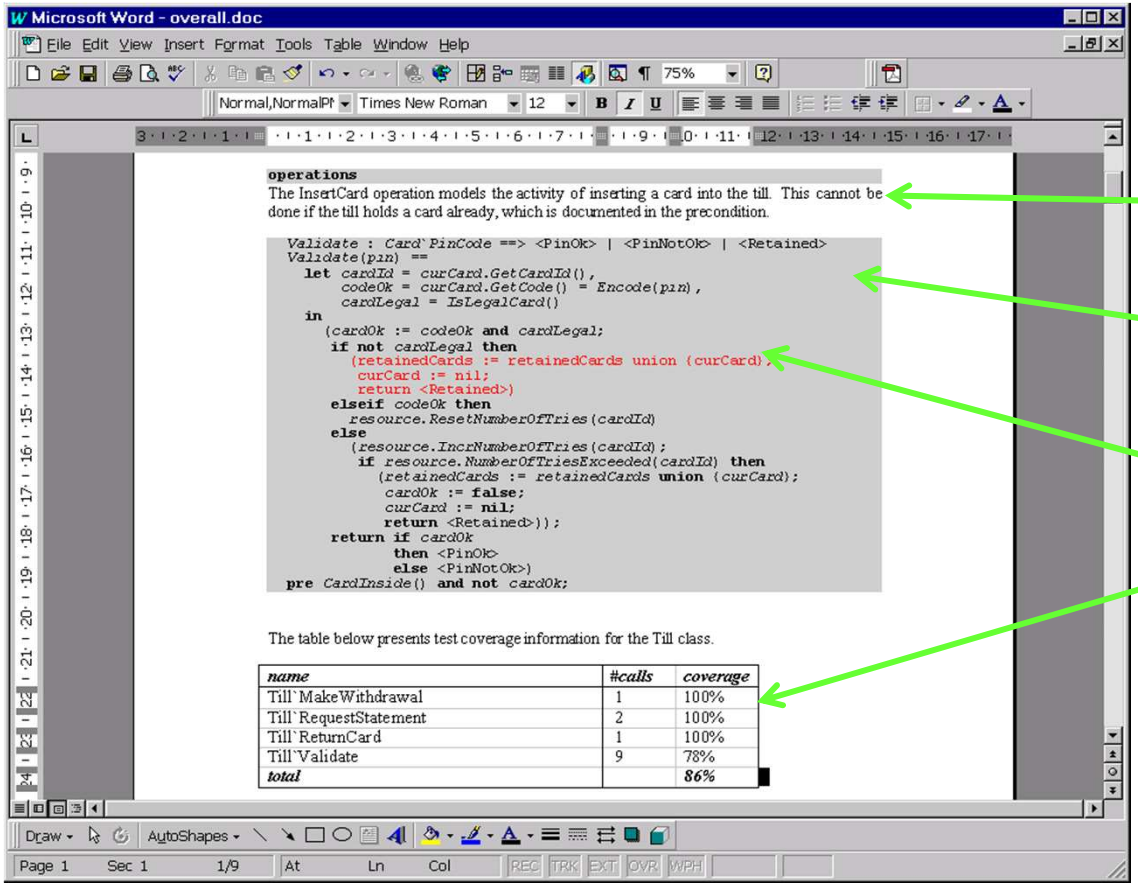


scope-complete:
all cases within small scope

Programa

- 3. Especificação baseada em modelos
  - As linguagens VDM-SL e VDM++.
  - Representação de dados com base em estruturas matemáticas.
  - Especificação com estado e sem estado.
  - Definição de tipos, valores e funções.
  - Definição de classes, variáveis de instância e operações.
  - Expressões e instruções.
  - ***Design-by-contract***: invariantes, pré-condições e pós-condições.
  - Descrição de algoritmos, especificações executáveis.
  - **Consistência da especificação**: obrigações de prova e teste.
  - Ligação do VDM++ ao UML.
  - Geração de código a partir de uma especificação formal.
- A ferramenta VDMTools.

Programa



One compound document:

- **Documentation**
- **Specification**
- **Test coverage**
- **Statistics**

Programa

- 4. Especificações baseadas em modelos (OCL)
  - A linguagem OCL
  - Comparação com VDM++

Programa

- 5. Lógica e "Model Checking”
  - Lógica proposicional, de predicados, temporal linear (LTL), temporal ramificada (CTL).
  - Representação de estados.
  - "Model Checking":
    - Propriedades: "Safety"; "Fairness"; "Liveness"; Universalidade; Inevitabilidade; Possibilidade; Ausência; Resposta; Precedência.
    - Problema de Explosão de estados (técnicas existentes para minorar este problema): Estado Simbólico; "Bounds"; "On-the-fly"; "Partial Order Reduction (POR)"; Abstracção.

Programa

- 6. Provas formais
  - Aplicação da lógica de Hoare à prova de correcção de algoritmos.
  - Linguagem de especificação Gallina: tipos e expressões; proposições e provas; tipos de dados indutivos; tácticas de prova e automação; predicados indutivos.
  - Prova de correcção de programas.

- As ferramentas Coq, Jape, Caduceus (C) e krakatoa (Java).

Avaliação do trabalho de VDM++

- 1. Contractos
  - 1.1. Pré-Condições
  - 1.2. Pós-Condições
  - 1.3. Invariantes
- 2. Mecanismos da Linguagem
  - 2.1. Funções e Operações
  - 2.2. Funções parciais
  - 2.3. Maps, Sets e Seqs
  - 2.4. Record Types, Tokens,...
- 3. Testes
  - 3.1. Cobertura dos Testes
  - 3.2. Significância dos Testes (testes a passar e testes a falhar)
- 4. Relatório
  - 4.1. Diagramas de UML/conversão para VDM++
  - 4.2. Restrições
  - 4.3. Matriz de Rastreabilidade
  - 4.4. Análise de Consistência
- 5. Apresentação/discussão

Matriz de rastreabilidade (exemplo)

Requirement Identifiers	Reqs Tested	REQ1 UC 1.1	REQ1 UC 1.2	REQ1 UC 1.3	REQ1 UC 2.1	REQ1 UC 2.2	REQ1 UC 2.3.1	REQ1 UC 2.3.2	REQ1 UC 2.3.3	REQ1 UC 2.4	REQ1 UC 3.1	REQ1 UC 3.2
Test Cases	321	3	2	3	1	1	1	1	1	1	2	3
1.1.1	1	x										
1.1.2	2		x	x								
1.1.3	2	x									x	
1.1.4	1			x								
1.1.5	2	x										x
1.1.6	1		x									
1.1.7	1			x								
1.2.1	2				x		x					
1.2.2	2					x		x				
1.2.3	2								x	x		
1.3.1	1										x	
etc…												
5.6.2	1										x	

Página da disciplina

- http://paginas.fe.up.pt/~apaiva/teach/1314/MFES.htm
  - Consultar os *links*:
    - Plano das aulas;
    - SiFEUP
    - Avaliação
    - Bibliografia
    - *Links* úteis