

MAC0340 - Laboratório de Engenharia de Software - 2014/S1

Antônio Augusto Tavares Martins Miranda (7644342)
amartmiranda@gmail.com

Rodrigo Duarte Louro (7240216)
digao.louro@gmail.com

Exercício 6: Avaliação da Adequação de testes

- **Ferramentas auxiliares para geração ou análise dos testes**

Para a geração dos testes nós não usamos nenhuma ferramenta, ou seja, criamos todos os nossos testes manualmente. Para executar os testes usamos a ferramenta JUnit.

- **Geração dos test cases para a análise da cobertura do MC/DC**

Tal como foi dito no tópico anterior, as malhas de testes, usadas no processo de determinação do grau de cobertura do MC/DC, foram elaboradas manualmente, tendo em conta o código fonte alvo e o grau de MC/DC que queríamos cobrir. O diretório `Atividade6/src/test/java/malhaDeTestesASerAnalizada` possui todas as malhas de teste elaboramos para este protótipo. As malhas de testes que se encontram neste diretório são:

- `ExemploClasseUmTeste`: esta classe de teste é usada para exercitar a classe `ExemploClasseUm` em relação aos requisitos do MC/DC da mesma. Dentro dele elaboramos dois testes com 100% de cobertura para o MC/DC.
- `ExemploClasseDoisTeste`: esta classe de teste é usada para exercitar a classe `ExemploClasseDois` em relação aos requisitos do MC/DC da mesma. Dentro dele elaboramos dois testes com 100% de cobertura para o MC/DC.
- `ExemploClasseTresTeste`: esta classe de teste é usada para exercitar a classe `ExemploClasseTres` em relação aos requisitos do MC/DC da mesma. Dentro dele elaboramos dois testes. Um com 100% de cobertura para o MC/DC e outro que nunca vai cobrir 100% do MC/DC.

- **Análise da Adequação da malha de testes**

Para fazer a avaliação das malhas de teste, estamos aproveitando o xml que gerado pelo protótipo do exercício 4, que contém os requisitos de teste para o cobrimento do MC/DC, e compará-los aos valores observados nas variáveis das decisões/condições do código fonte, resultantes da execução da malha de testes.

Primeiramente extraímos os requisitos de teste para o MC/DC do código fonte, presentes no xml gerado pelo programa do exercício 4, para uma hashtable de decisões/condição. Em seguida implantamos diretamente no código fonte (do programa a ser testado), operações de captura e armazenamento dos valores das variáveis essenciais, particulares a cada decisão/condição do programa e geramos (manualmente) as malhas de teste para exercitar o código fonte. Após a execução das malhas de teste é criada uma hashtable com as valorações de cada uma das variáveis essenciais às decisões/condições do programa e que foram exercitadas pela malha de teste. Para computar o grau de cobrimento do mc/dc comparamos as valorações da hashtable com os requisitos para o cobrimento do mc/dc com a hashtable dos valores exercitados pela malha de testes.

- **Manual de usuário**

...

- **Testes para a validação do protótipo**

Para a validação do nosso protótipo criamos duas classes de testes. A `HashTest`, visto todas as funcionalidades mais importantes do mesmo se encontram implementadas na classe `Hash`. Esta classe contém as implementações da montagem das hashes como as valores da execução das malhas de teste e os valores necessários para a cobertura do mc/dc, comparação das hashes com os valores resultantes da execução das malhas de teste e os valores necessários para o cobrimento do mc/dc e por fim a implementação da determinação de cobertura do mc/dc de cada uma da

classes e a sua respectiva impressão. A `LeituraXMLTest` é simplesmente para garantir que as condições do mcdc estão sendo lidas corretamente do arquivo xml.

Na `HashTest` temos os seguintes testes:

- `setAndGetHashExecutadosTest`: este método testa os getters e setters da `Hashtable` que contem as valorações da execução das malhas de teste;
- `comparaHashTablesTest`: este método testa a comparação da `hashtable` dos valores exercitados e a `hashtable` dos valores para o cobrimento do mcdc para a determinação da porcentagem de cobertura do mcdc.

Na `LeituraXMLTest` temos o seguinte teste:

- `getRequisitosMCDCTest`: este método testa se a extração dos requisitos do mcdc está sendo feita corretamente.

- **Conclusões**

...