|  |  |
| --- | --- |
|  | *Code Inspection Report*  *Anti-Spam Configuration Software Development Project*  BSc/MSc in [LEI | LIGE | METI]  Academic Year 2017/2018 - 1º Semester  Software Engineering I  Group 44  83169, João Lola, METIA1  81611, Délcio Pedro, METI-PL  79170, Ansumane Mané, METI-PL  83455, Cedrick Beta, METIA1  ISCTE-IUL, Instituto Universitário de Lisboa  1649-026 Lisbon  Portugal  November 25th 2017 |

**Table of Contents**

[Introduction 1](#_Toc501710336)

[Code inspection – Leitura e Escrita no Ficheiro 2](#_Toc501710337)

[Code inspection – Gerar pesos aleatórios para atribuição as regras 2](#_Toc501710338)

[Code inspection – Cálculo de Falsos Positivos e Negativos 3](#_Toc501710339)

[Code inspection – Gerar pesos otimizados 3](#_Toc501710340)

[Code inspection – Checklist 4](#_Toc501710341)

[Found defects 7](#_Toc501710342)

[Corrective measures 8](#_Toc501710343)

[Conclusions of the inspection process 8](#_Toc501710344)

# Introduction

O software produzido neste projeto consiste num filtro de configuração antisspam destinado a uma caixa de email de lazer. O objetivo neste *software* era permitir gerar uma configuração aleatória por um utilizador para classificação de mensagens ilegítimas(*Spam*) ou legitimas(*Ham*), permitindo lhe substituir pesos calculados aleatoriamente para cada uma das regras carregadas a partir de um ficheiro. Para disso, o utilizador poderia também gerar uma configuração de maneira automática recorrendo a um algoritmo de otimização chamado NSGA-II disponibilizado na *framework* de otimização *jMetal* 4.5.2, destinada a otimização recorrendo a meta-heurísticas. Com base na configuração gerada manualmente e automaticamente, o filtro deveria avaliar a qualidade da configuração tanto gerada aleatoriamente como gerada automaticamente com o objetivo de obter a melhor configuração para a caixa de correio, em questão.

# Code inspection – Leitura e Escrita no Ficheiro

|  |  |
| --- | --- |
| *Meeting date:*  *Meeting duration:*  *Moderator:*  *Producer:*  *Inspector:*  *Recorder:* | *29/11/2017*  *90 minutes*  *João Lola*  *Cedrick Beta*  *Délcio Pedro*  *Ansumane Mané* |
| *Component name (Package/Class/Method):* | *(pt.iscteiul.antispamfilter.models.dao/DadosDao.java/leituraFicheiro)*  *(pt.iscteiul.antispamfilter.models.dao/DadosDao.java/escritaFicheiro)* |
| *Component was compiled:* | *Sim* |
| *Component was executed:* | *Sim* |
| *Component was tested without errors:* | *Sim* |
| *Testing coverage achieved:* | *100%* |

# Code inspection – Gerar pesos aleatórios para atribuição as regras

|  |  |
| --- | --- |
| *Meeting date:*  *Meeting duration:*  *Moderator:*  *Producer:*  *Inspector:*  *Recorder:* | *01/12/2017*  *30 minutes*  *João Lola*  *Cedrick Beta*  *Délcio Pedro*  *Ansumane Mané* |
| *Component name (Package/Class/Method):* | *(pt.iscteiul.antispamfilter.models/AntiSpamMethods/gerarPesos)* |
| *Component was compiled:* | *Sim* |
| *Component was executed:* | *Sim* |
| *Component was tested without errors:* | *Sim* |
| *Testing coverage achieved:* | *100%* |

# Code inspection – Cálculo de Falsos Positivos e Negativos

|  |  |
| --- | --- |
| *Meeting date:*  *Meeting duration:*  *Moderator:*  *Producer:*  *Inspector:*  *Recorder:* | *08/12/2017*  *120 minutes*  *João Lola*  *Ansumane Mané*  *Délcio Pedro*  *Cedrick Beta* |
| *Component name (Package/Class/Method):* | *(pt.iscteiul.antispamfilter.models/AntiSpamMethods.java/calcularFpEFn)* |
| *Component was compiled:* | *Sim* |
| *Component was executed:* | *Sim* |
| *Component was tested without errors:* | *Sim* |
| *Testing coverage achieved:* | *83%* |

# Code inspection – Gerar pesos otimizados

|  |  |
| --- | --- |
| *Meeting date:*  *Meeting duration:*  *Moderator:*  *Producer:*  *Inspector:*  *Recorder:* | *15/12/2017*  *120 minutes*  *João Lola*  *Délcio Pedro*  *Ansumane Mané*  *Cedrick Beta* |
| *Component name (Package/Class/Method):* | *(pt.iscteiul.antispamfilter.models/AntiSpamMethods.java/gerarPesosOtimizados)* |
| *Component was compiled:* | *Sim* |
| *Component was executed:* | *Sim* |
| *Component was tested without errors:* | *Sim* |
| *Testing coverage achieved:* | *100%* |

# Code inspection – Checklist

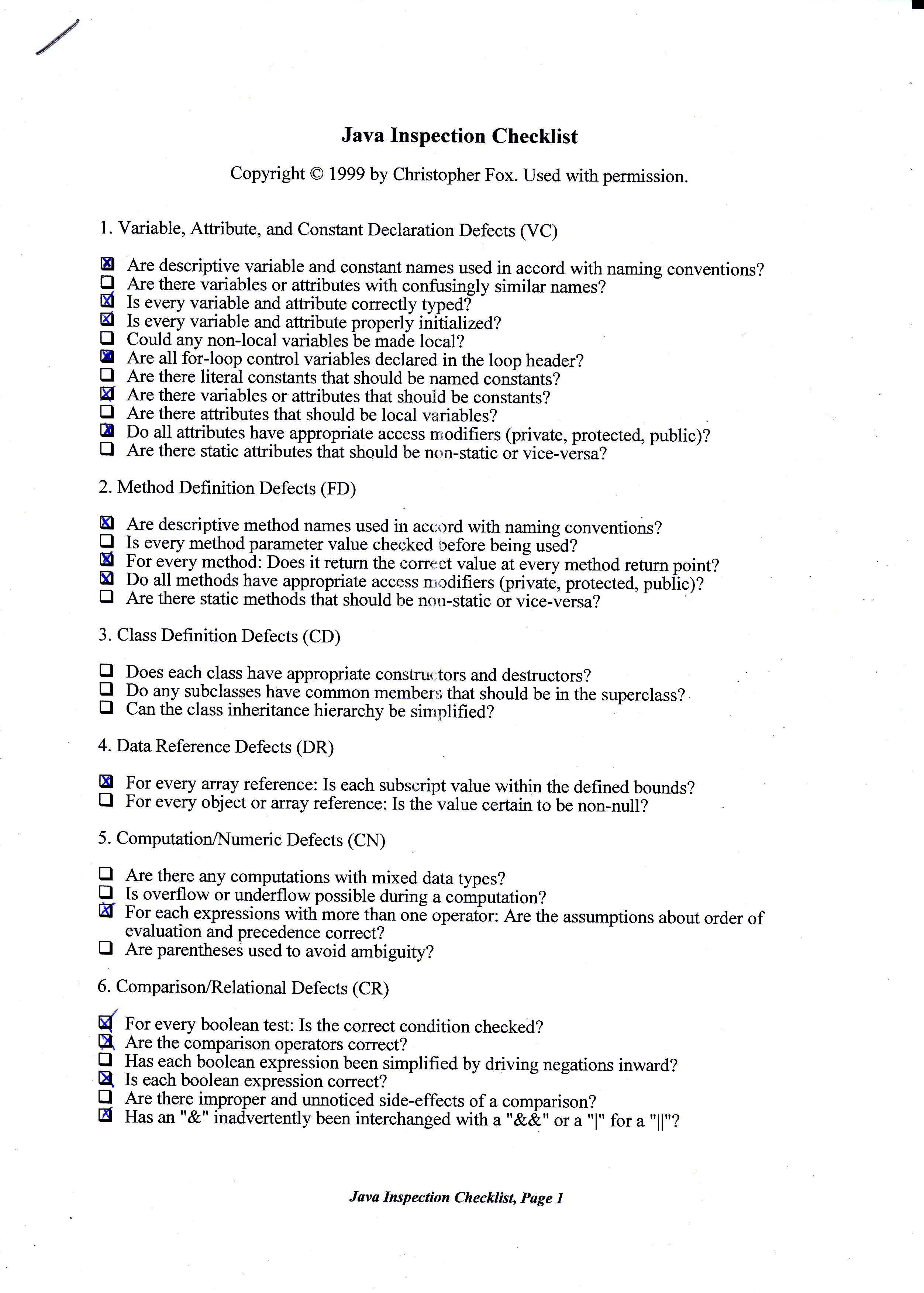


Figure 1 - Code Inspection Check List – Página 1/3



Figure 2 - Code Inspectino Check List - Pagina 2/3

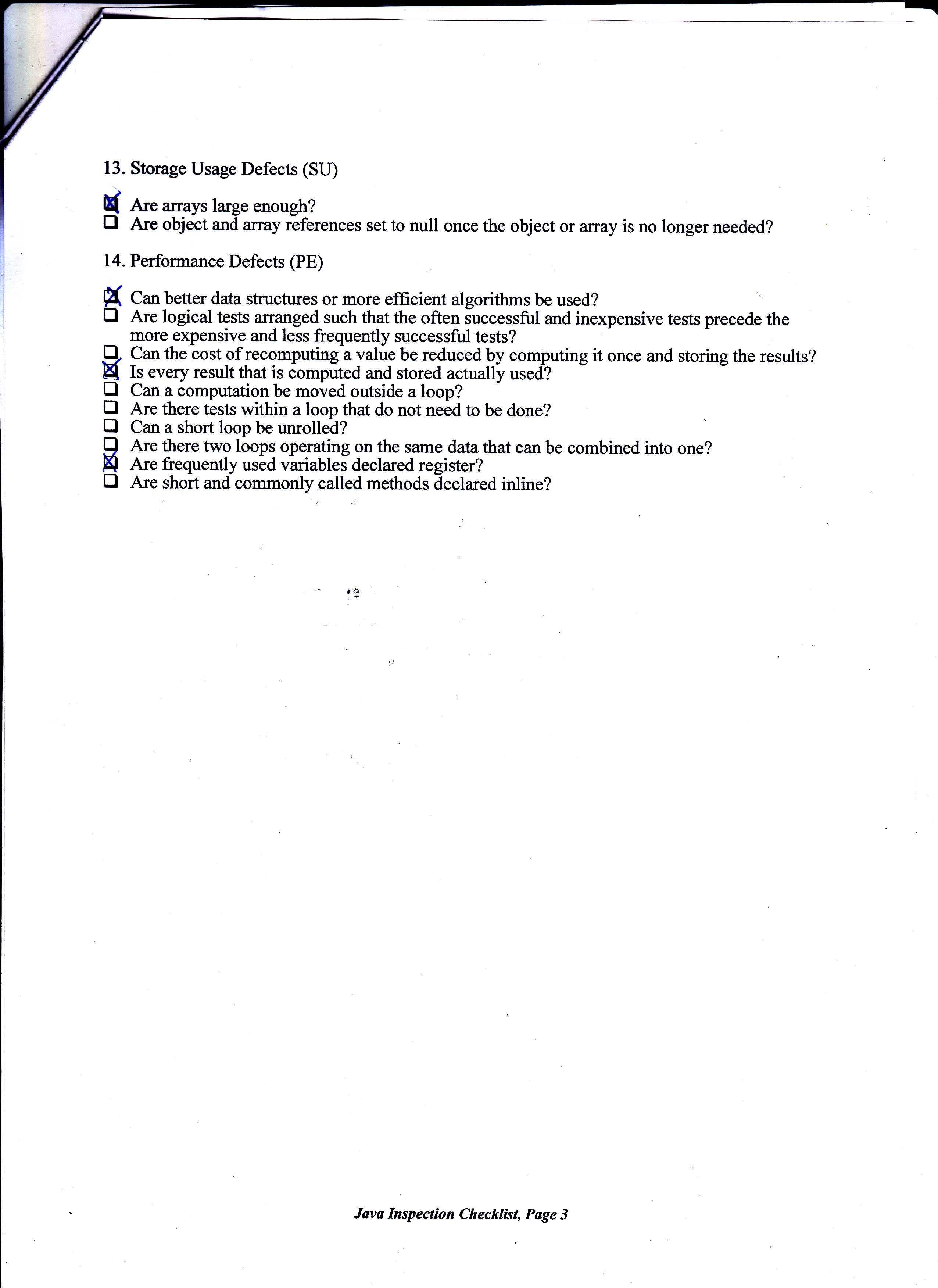


Figure 3 - Code Inspection Check List - Página 3/3

# Found defects

Identify and describe found defects, opinions and suggestions.

|  |  |  |  |
| --- | --- | --- | --- |
| **Found defect Id** | **Package, Class, Method, Line** | **Defect category** | **Description** |
| 1 | Package: pt.iscteiul.antispamfilter.models  Class:  AntiSpamMethods.java  Method:  gerarPesos()  Linha:  23-24 | Pequeno | O atributo das variáveis pesoMin e pesoMax devia ser final, porque são consideradas constantes que definem o intervalo de números para gerar valores. |
| 2 | Package:  pt.iscteiul.antispamfilter.controller  Class:  AntiSpamFilterController.java  Method:  overwriteSavedConfiguration()  line:  181 | Pequeno | Era escusado utilizar o *printwriter* para guardar os pesos dentro do ficheiro rules.cf, dando que foi criado um método chamado escreverFicheiro dentro da classe DadosDao.java. |
| 3 | Package:  pt.iscteiul.antispamfilter.models  Class:  AntispamfilterMethods.java  Method  calcularFpEFn()  line:  66 | Pequeno | Não seria necessário recorrer a incrementação do vetor fx, dado que se faz a incrementação do falso positivo e falso negativo recorrendo a variáveis para esse efeito. |

# Corrective measures

|  |  |  |  |
| --- | --- | --- | --- |
| **Found defect Id** | **How** | **When** | **Who** |
| 1 | A correção deste defeito passaria por declarar as variáveis pesoMin e pesoMax como final, impossibilitando o valor de ser alterado durante a execução. | Este defeito deverá ser corrigido numa próxima versão do mesmo ou numa próxima incrementação para implementar um conjunto adicional de funcionalidades. | Este defeito deverá ser corrigido pelo programador responsável pela implementação desta funcionalidade. |
| 2 | A correção deste defeito passaria por instanciar a classe DadosDao.java chamado o método escreverFicheiro passando lhe qual o ficheiro que este deve abrir e o conteúdo que deve introduzir dentro do ficheiro. | Este defeito deverá ser corrigido numa próxima versão do mesmo ou numa próxima incrementação para implementar um conjunto adicional de funcionalidades. | Este defeito deverá ser corrigido pelo programador responsável pela implementação desta funcionalidade. |
| 3 | Bastaria retirar o vetor fx, pois a incrementação poderia ser realizada pelas variáveis falsoPositivo, quando o valor é maior ou igual a 5 e falsoNegativo., quando o valor é menor ou igual 5. | Este defeito deverá ser corrigido numa próxima versão do mesmo ou numa próxima incrementação para implementar um conjunto adicional de funcionalidades. | Este defeito deverá ser corrigido pelo programador responsável pela implementação desta funcionalidade. |

# Conclusions of the inspection process

Em suma, o código que foi desenvolvido no âmbito deste projeto precisa de facto de algumas correções mínimas tais como alterar o tipo de uma variável e alteração de algum método, o que o tornaria mais legível e otimizado. Apesar destas correções a serem realizadas e não termos conseguido desativar a inspeção de código nas classes não desenvolvidas por nós verificou-se que foi possível atingir a taxa mínima de cobertura de 76% para a métrica *complexity*.