Relatório Part.2

Engenharia de Software II

Gabriel Pereira 8150115

Nuno Josefino 8150189

Rafael Vieira 8150448



# Índice

[Índice 2](#_Toc532769049)

[Introdução 5](#_Toc532769050)

[Âmbito do Projeto 6](#_Toc532769051)

[No âmbito 6](#_Toc532769052)

[Fora do âmbito 6](#_Toc532769053)

[Metódo removeDigitos 7](#_Toc532769054)

[Tabela ECP 7](#_Toc532769055)

[Tabela BVA 7](#_Toc532769056)

[Casos de teste 7](#_Toc532769057)

[Caso de teste 1 () 7](#_Toc532769058)

[Caso de teste 2 () 7](#_Toc532769059)

[Caso de teste 3 () 7](#_Toc532769060)

[Caso de teste 4 () 7](#_Toc532769061)

[Caso de teste 5 () 7](#_Toc532769062)

[Metódo removeCaracteresPont 8](#_Toc532769063)

[Tabela ECP 8](#_Toc532769064)

[Tabela BVA 8](#_Toc532769065)

[Caso de teste 8](#_Toc532769066)

[Caso de teste 1 () 8](#_Toc532769067)

[Caso de teste 2 () 8](#_Toc532769068)

[Caso de teste 3 () 8](#_Toc532769069)

[Caso de teste 4 () 8](#_Toc532769070)

[Caso de teste 5 () 8](#_Toc532769071)

[Caso de teste 6 () 8](#_Toc532769072)

[Metódo matrizOcorrencias 9](#_Toc532769073)

[Tabela ECP 9](#_Toc532769074)

[Tabela BVA 9](#_Toc532769075)

[Caso de teste 9](#_Toc532769076)

[Caso de teste 1 () 9](#_Toc532769077)

[Caso de teste 2 () 9](#_Toc532769078)

[Caso de teste 3 () 9](#_Toc532769079)

[Caso de teste 4 () 9](#_Toc532769080)

[Caso de teste 5 () 9](#_Toc532769081)

[Caso de teste 6 () 9](#_Toc532769082)

[Caso de teste 7 () 9](#_Toc532769083)

[Caso de teste 8 () 9](#_Toc532769084)

[Caso de teste 9 () 9](#_Toc532769085)

[Metódo matrizQuery 10](#_Toc532769086)

[Tabela ECP 10](#_Toc532769087)

[Tabela BVA 10](#_Toc532769088)

[Caso de teste 10](#_Toc532769089)

[Caso de teste 1 () 10](#_Toc532769090)

[Caso de teste 2 () 10](#_Toc532769091)

[Caso de teste 3 () 10](#_Toc532769092)

[Metódo verificarSemelhanca 11](#_Toc532769093)

[Tabela ECP 11](#_Toc532769094)

[Tabela BVA 11](#_Toc532769095)

[Caso de teste 11](#_Toc532769096)

[Caso de teste 1 () 11](#_Toc532769097)

[Caso de teste 2 () 11](#_Toc532769098)

[Caso de teste 3 () 11](#_Toc532769099)

[Caso de teste 4 () 11](#_Toc532769100)

[Caso de teste 5 () 11](#_Toc532769101)

[Metódo listaSim 12](#_Toc532769102)

[Tabela ECP 12](#_Toc532769103)

[Tabela BVA 12](#_Toc532769104)

[Caso de teste 12](#_Toc532769105)

[Caso de teste 1 () 12](#_Toc532769106)

[Caso de teste 2 () 12](#_Toc532769107)

[Caso de teste 3 () 12](#_Toc532769108)

[Caso de teste 4 () 12](#_Toc532769109)

[Caso de teste 5 () 12](#_Toc532769110)

[Caso de teste 6 () 12](#_Toc532769111)

[Caso de teste 7 () 12](#_Toc532769112)

[Caso de teste 8 () 12](#_Toc532769113)

[Metódo listaSimGrau 13](#_Toc532769114)

[Tabela ECP 13](#_Toc532769115)

[Tabela BVA 13](#_Toc532769116)

[Caso de teste 13](#_Toc532769117)

[Caso de teste 1 () 13](#_Toc532769118)

[Caso de teste 2 () 13](#_Toc532769119)

[Caso de teste 3 () 13](#_Toc532769120)

[Caso de teste 4 () 13](#_Toc532769121)

[Caso de teste 5 () 13](#_Toc532769122)

[Caso de teste 6 () 13](#_Toc532769123)

[Caso de teste 7 () 13](#_Toc532769124)

[Caso de teste 8 () 13](#_Toc532769125)

[Metódo listaSimMax 14](#_Toc532769126)

[Tabela ECP 14](#_Toc532769127)

[Tabela BVA 14](#_Toc532769128)

[Caso de teste 14](#_Toc532769129)

[Caso de teste 1 () 14](#_Toc532769130)

[Caso de teste 2 () 14](#_Toc532769131)

[Caso de teste 3 () 14](#_Toc532769132)

[Caso de teste 4 () 14](#_Toc532769133)

[Caso de teste 5 () 14](#_Toc532769134)

[Caso de teste 6 () 14](#_Toc532769135)

[Caso de teste 7 () 14](#_Toc532769136)

[Caso de teste 8 () 14](#_Toc532769137)

[Conclusão 15](#_Toc532769138)

# Introdução

O presente trabalho surge no âmbito da Unidade Curricular de Engenharia de Software II e em especifico da parte pratica da mesma.

Foi-nos apresentado um desafio no dominio da engenharia de software e em particular na áre da qualidade de software.

Desafio este que consiste em desenvolver um motor de pesquisa que através da introdução de uma frase na query, infere sobre um repositório de ficheiros de texto, sendo necessário o desenvolvimento de uma bibilioteca de software capaz de processar coleções de ficheiros de modo a encontrar ficheiro com um grau de similariedade elevado através de uma query de pesquisa.

Esta parte inicial tem como objetivo a especificação de casos de teste utilizando as técnicas *Equivalence Class Partitioning* e *Boundary Value Analysis* no nível de testes unitários especificando casos de teste para cada método. Deverá ser descrito igualmente *test inputs*, *execution conditions* e *expected outputs*, assegurando que os casos de teste cobrem *Valid equivalence classes* e *Invalid equivalence classes*.

Para elaboração deste projeto, pretendemos utilizar ferramentas com o Github para a hospedagem de todos os ficheiros do nosso projeto e o Youtrack para gerenciar todos os passos a ser tomados no nosso projeto, desde o que falta concretizar ao que irá ser elaborado e se tal irá ocorrer dentro dos prazos a ser estimados.

Prentende-se ainda realizar um estudo das restantes ferramentas a serem utilizadas por nós durante o decorrer do projeto.

Como último objetivo define-se a elaboração de um relatório de testes.

# Âmbito do Projeto

## No âmbito

No âmbito desta fase do trabalho inclui-se a definição de requisitos, realização de tabelas *ECP*, *BVA*, diagrama de casos de uso e relatório de testes.

De igual forma, inclui-se a criação de um repositório *GIT* e a sua configuração, o estudo e criação de um projeto no *Youtrack* enquanto ferramenta de *issue tracking* para o desenvolvimento do projeto.

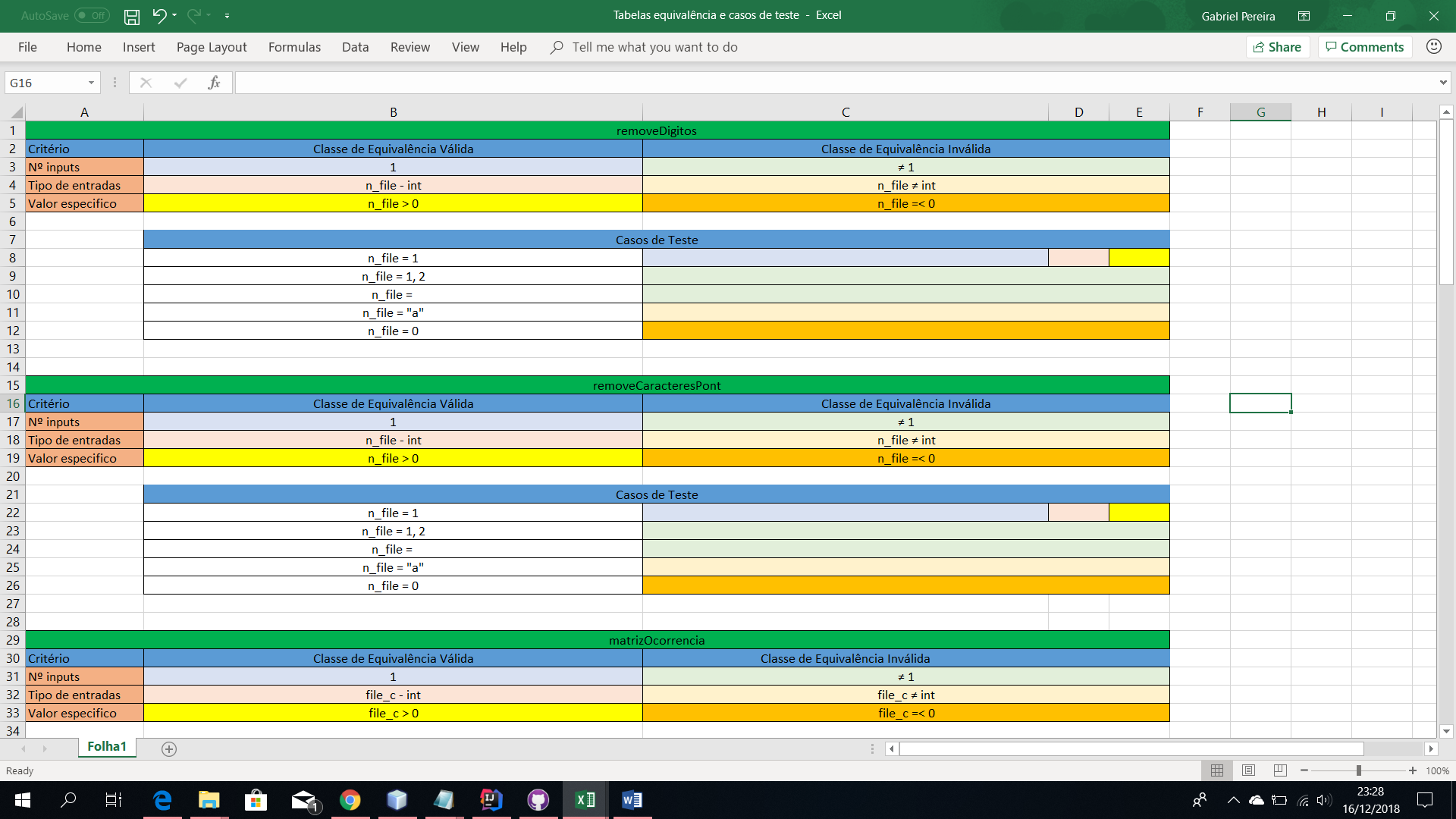
É incluído também o estudo das restantes ferramentas que irão ser configuradas e utilizadas nos sprints seguintes deste projeto.

## Fora do âmbito

Fora do âmbito desta fase do projeto, inclui-se desenvolvimento de software funcional bem como a utilização de ferramentas para codificação de testes, revisão de código, ferramentas para implementar os conceitos de *CI* durante o desenvolvimento do projeto como *builds* e testes automáticos ou linguagens de programação usada para o desenvolvimento do motor de pesquisa e especificação de testes.

# Metódo removeDigitos

## Tabela ECP



## Tabela BVA

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Método | removeDigitos | | | | |
|  |  |  |  |  |  |
| Cenários |  |  |  |  |  |
| 1 | n\_file > 0 | | | | |
| 2 | n\_file =< 0 | | | | |
|  |  |  |  |  |  |
| Casos de Teste | Input | | |  | Output |
|  | **n\_file** | | |  |  |
| 1 | **1** | | |  | Documentos sem digitos/caracteres especiais |
| 2 | **0** | | |  | -1 |

## Casos de teste

## Caso de teste 1 ()

## Caso de teste 2 ()

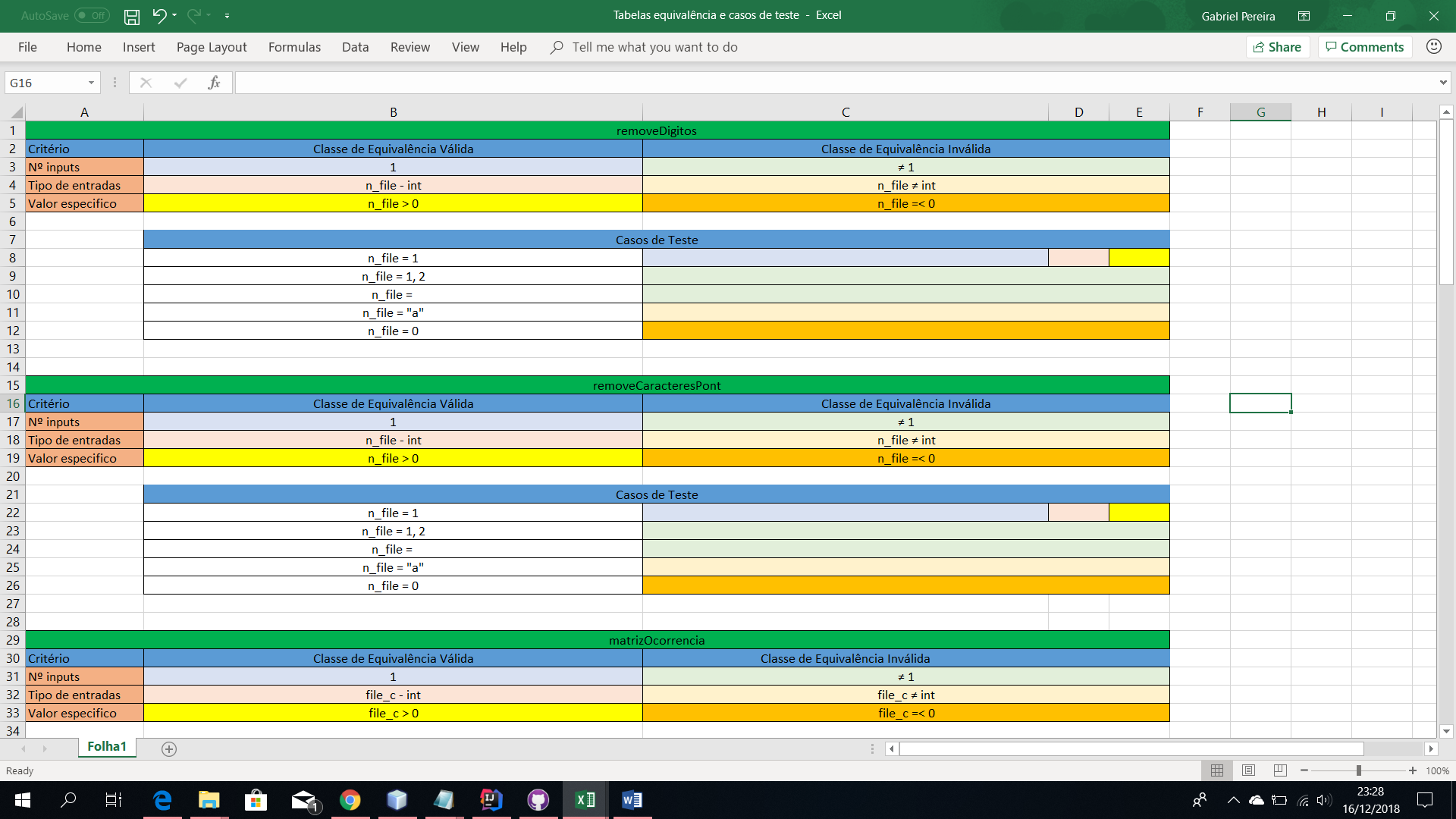
## Caso de teste 3 ()

## Caso de teste 4 ()

## Caso de teste 5 ()

# Metódo removeCaracteresPont

## Tabela ECP



## Tabela BVA

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Método | removeCaracteresPont | | | | |
|  |  |  |  |  |  |
| Cenários |  |  |  |  |  |
| 1 | n\_file > 0 | | | | |
| 2 | n\_file =< 0 | | | | |
|  |  |  |  |  |  |
| Casos de Teste | Input | | |  | Output |
|  | **n\_file** | | |  |  |
| 1 | **1** | | |  | Documentos sem digitos/caracteres especiais |
| 2 | **0** | | |  | -1 |

## Caso de teste

## Caso de teste 1 ()

## Caso de teste 2 ()

## Caso de teste 3 ()

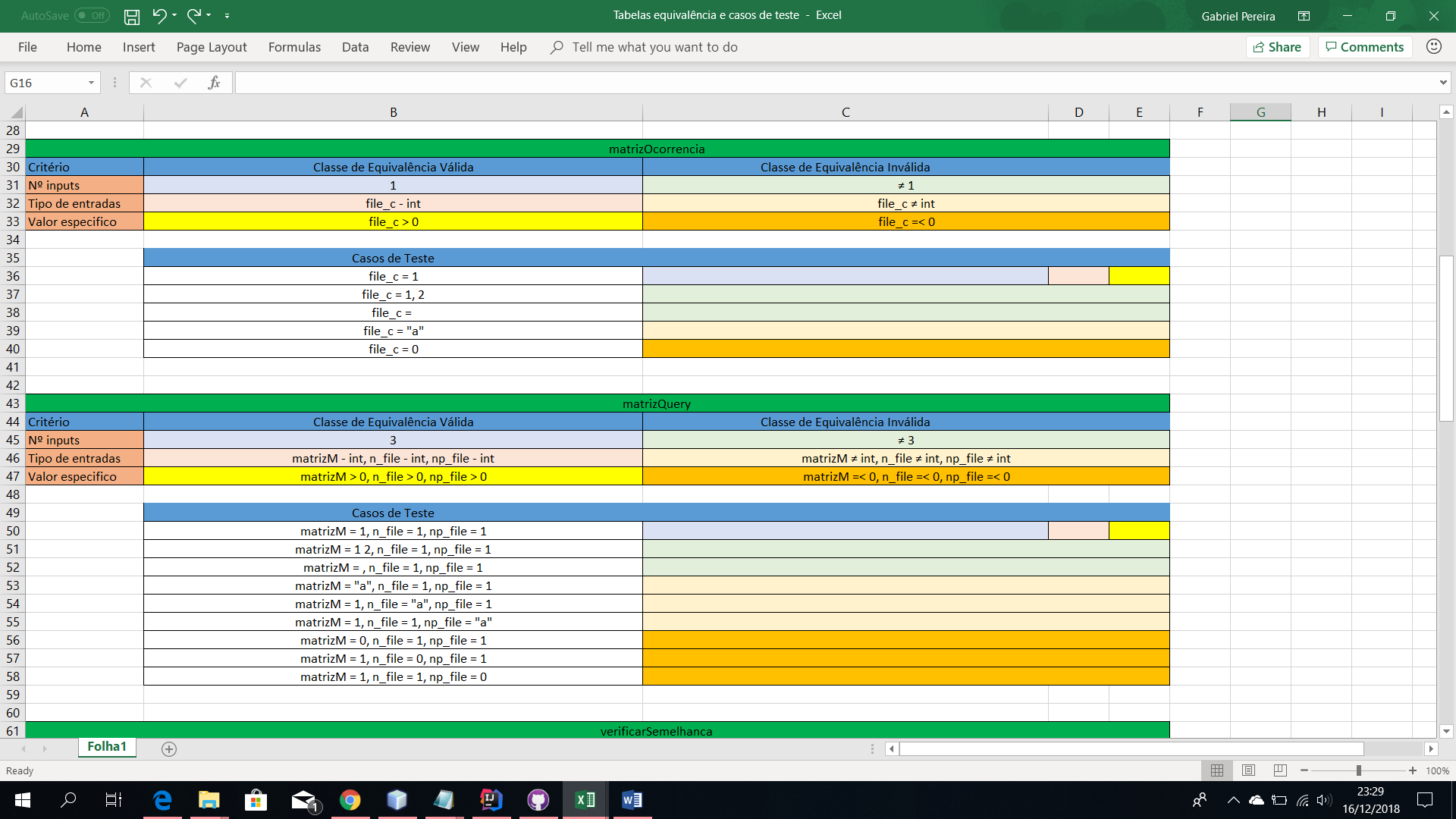
## Caso de teste 4 ()

## Caso de teste 5 ()

## Caso de teste 6 ()

# Metódo matrizOcorrencias

## Tabela ECP



## Tabela BVA

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Método | matrizOcorrencias | | | | |
|  |  |  |  |  |  |
| Cenários |  |  |  |  |  |
| 1 | c\_file > 0 | | | | |
| 2 | c\_file =< 0 | | | | |
|  |  |  |  |  |  |
| Casos de Teste | Input |  |  |  | Output |
|  | **c\_file** | | |  |  |
| 1 | **1** | | |  | Matriz M |
| 2 | **0** | | |  | -1 |
|  |  |  |  |  |  |

## Caso de teste

## Caso de teste 1 ()

## Caso de teste 2 ()

## Caso de teste 3 ()

## Caso de teste 4 ()

## Caso de teste 5 ()

## Caso de teste 6 ()

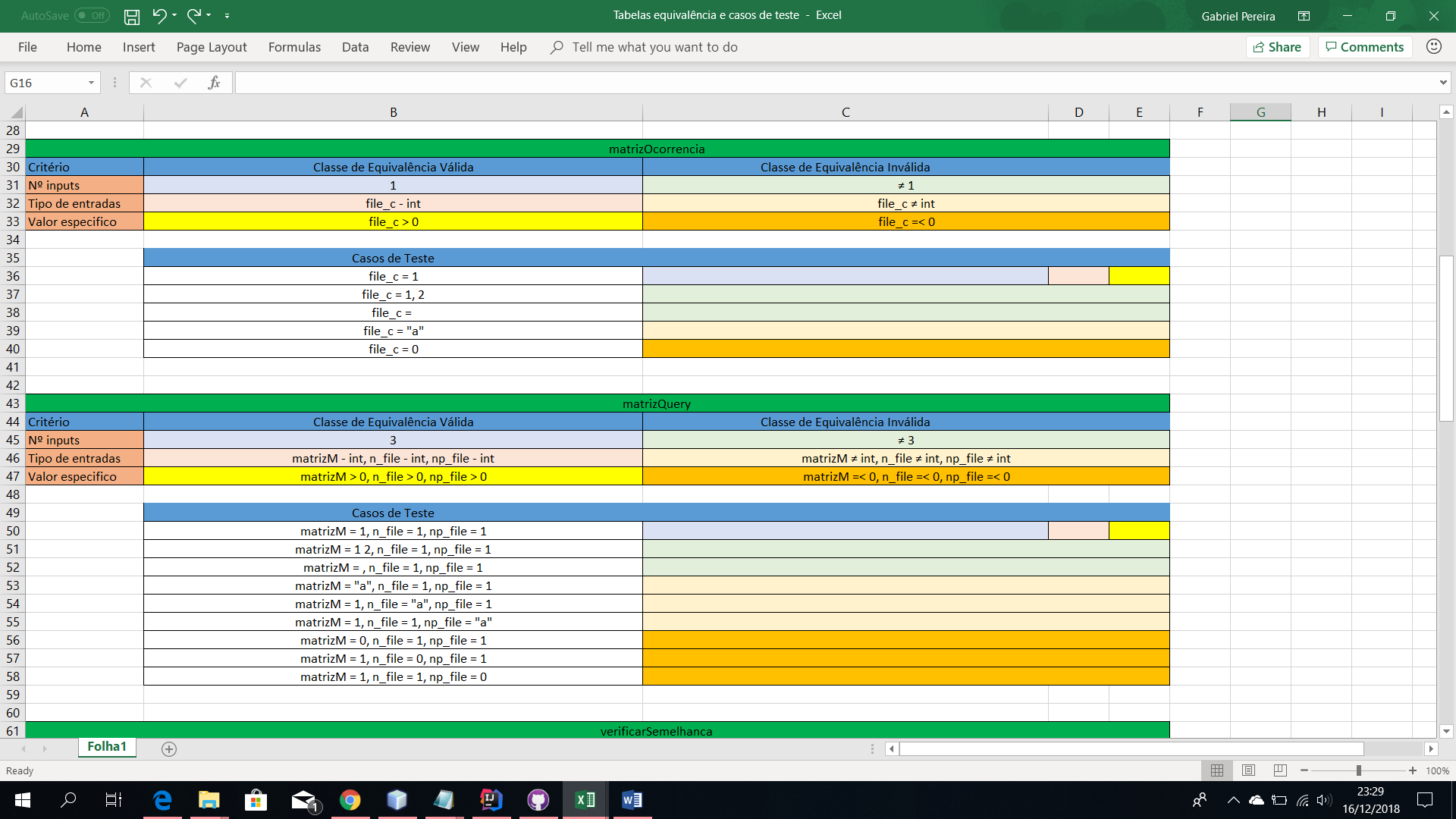
## Caso de teste 7 ()

## Caso de teste 8 ()

Caso de teste 9 ()

## Metódo matrizQuery

## Tabela ECP



## Tabela BVA

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Método | matrizQuery | | | | |
|  |  |  |  |  |  |
| Cenários |  |  |  |  |  |
| 1 | matrizM > 0, n\_file > 0, np\_file > 0 | | | | |
| 2 | matrizM =< 0, n\_file > 0, np\_file > 0 | | | | |
| 3 | matrizM > 0, n\_file =< 0, np\_file > 0 | | | | |
| 4 | matrizM > 0, n\_file > 0, np\_file =< 0 | | | | |
|  |  |  |  |  |  |
| Casos de Teste | Input | | |  | Output |
|  | **matrizM** | **n\_file** | **np\_file** |  |  |
| 1 | 1 | 1 | 1 |  | arrayQ |
| 2 | 0 | 1 | 1 |  | -1 |
| 3 | 1 | 1 | 1 |  | arrayQ |
| 4 | 1 | 0 | 1 |  | -1 |
| 5 | 1 | 1 | 1 |  | arrayQ |
| 6 | 1 | 1 | 0 |  | -1 |

## Caso de teste

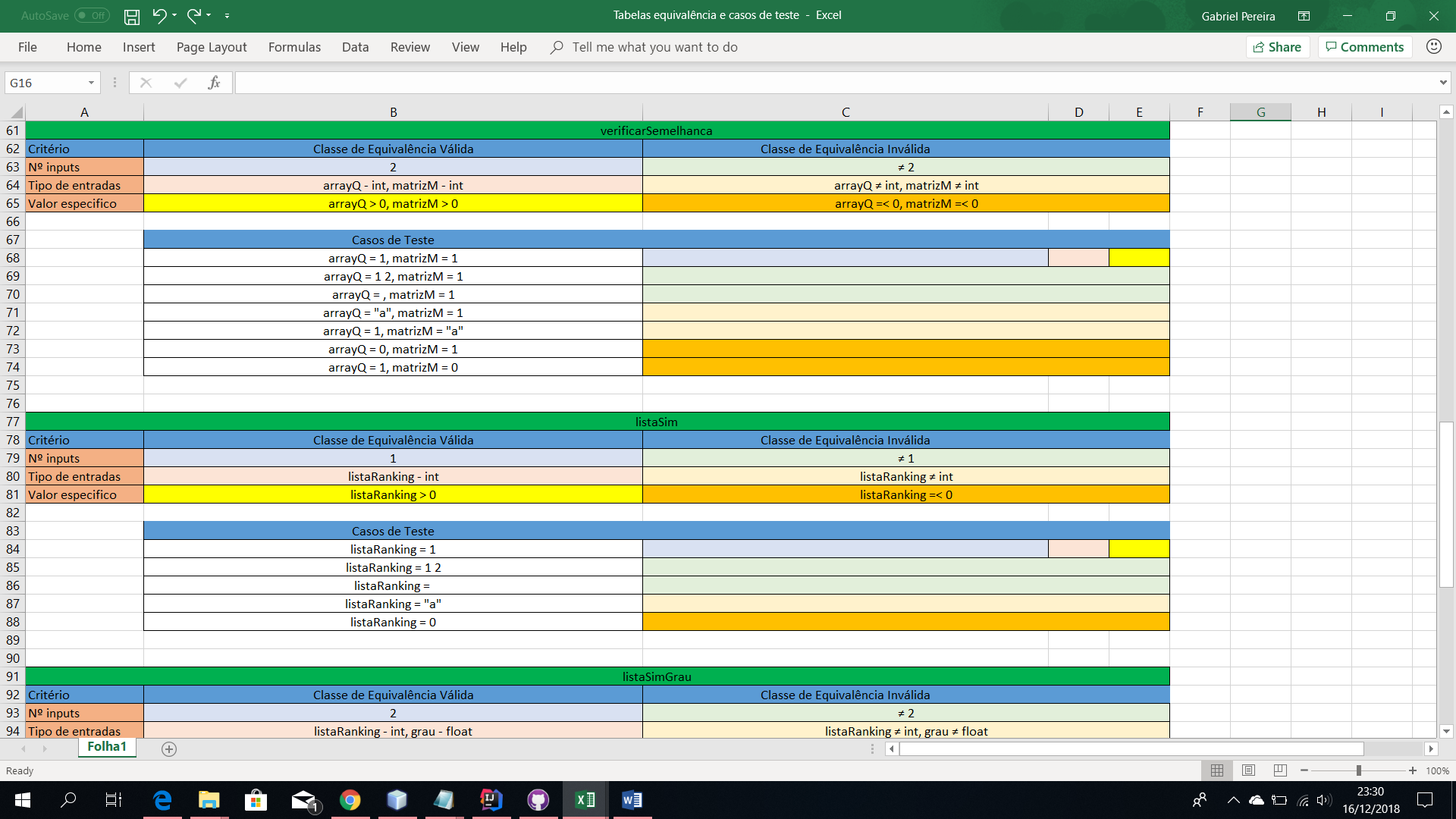
## Caso de teste 1 ()

## Caso de teste 2 ()

## Caso de teste 3 ()

# Metódo verificarSemelhanca

## Tabela ECP



## Tabela BVA

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Método | verificarSemelhanca | | | | |
|  |  |  |  |  |  |
| Cenários |  |  |  |  |  |
| 1 | arrayQ > 0, matrizM > 0 | | | | |
| 2 | arrayQ =< 0, matrizM > 0 | | | | |
| 2 | arrayQ > 0, matrizM =< 0 | | | | |
|  |  |  |  |  |  |
| Casos de Teste | Input | |  |  | Output |
|  | **arrayQ** | **matrizM** |  |  |  |
| 1 | 1 | 1 |  |  | grauSim |
| 2 | 0 | 1 |  |  | -1 |
| 3 | 1 | 1 |  |  | grauSim |
| 4 | 1 | 0 |  |  | -1 |

## Caso de teste

## Caso de teste 1 ()

## Caso de teste 2 ()

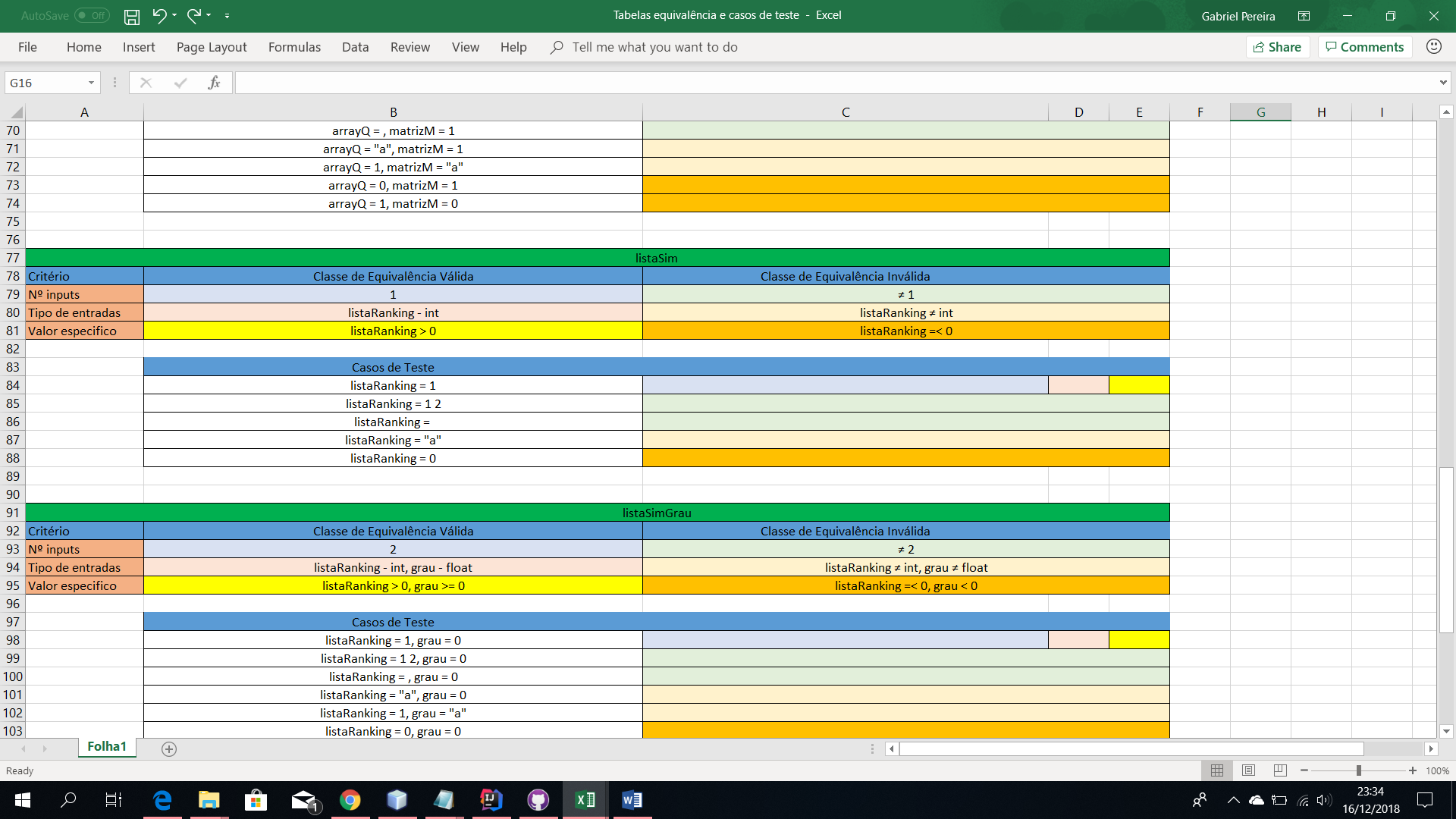
## Caso de teste 3 ()

## Caso de teste 4 ()

## Caso de teste 5 ()

# Metódo listaSim

## Tabela ECP



## Tabela BVA

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Método | listaSim | | | | |
|  |  |  |  |  |  |
| Cenários |  |  |  |  |  |
| 1 | listaRanking > 0 | | | | |
| 2 | listaRanking =< 0 | | | | |
|  |  |  |  |  |  |
| Casos de Teste | Input | |  |  | Output |
|  | **listaRanking** | |  |  |  |
| 1 | **1** |  |  |  | lista ordenada por similaridade |
| 2 | **0** |  |  |  | -1 |

## Caso de teste

## Caso de teste 1 ()

## Caso de teste 2 ()

## Caso de teste 3 ()

## Caso de teste 4 ()

## Caso de teste 5 ()

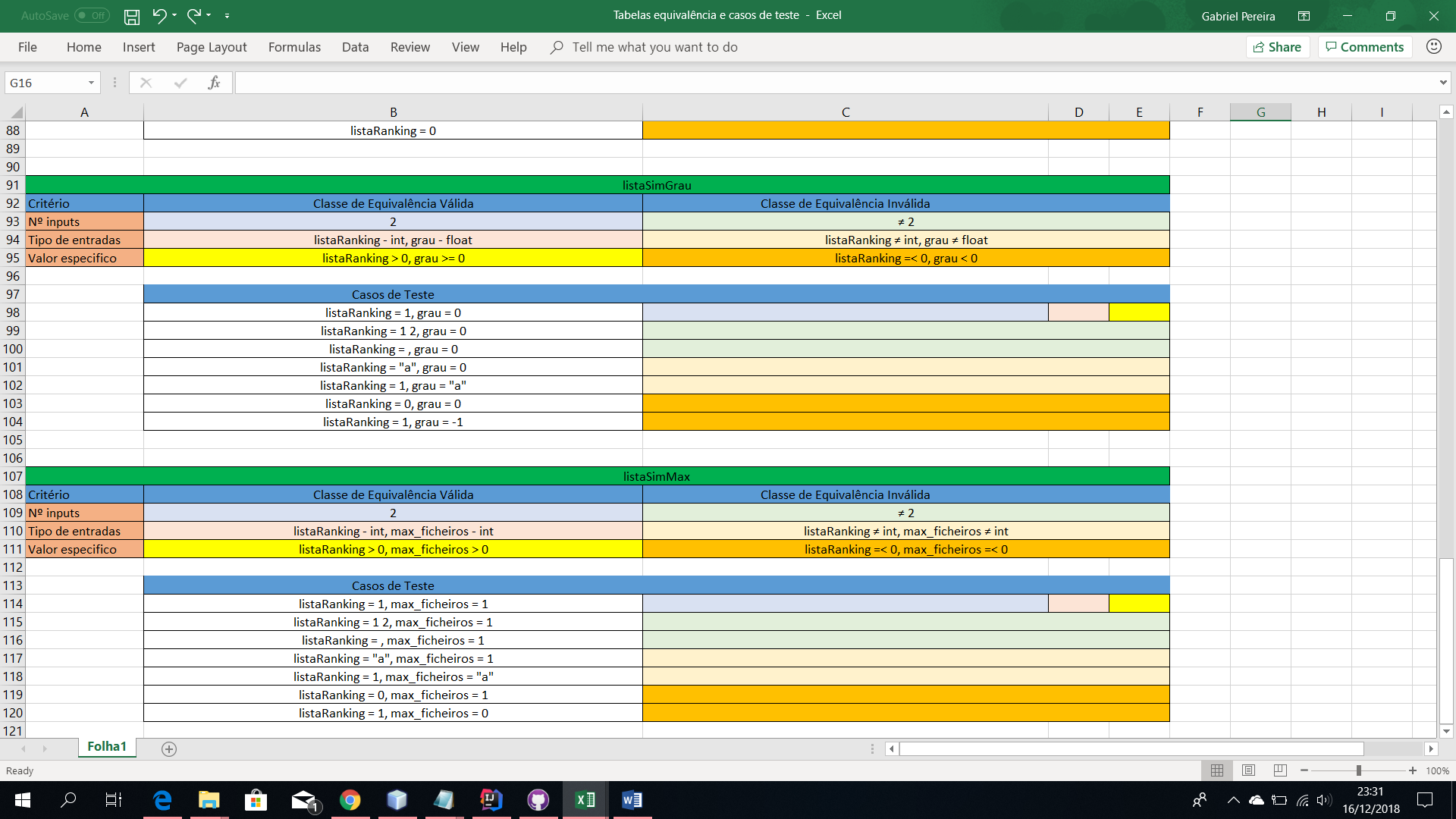
## Caso de teste 6 ()

## Caso de teste 7 ()

## Caso de teste 8 ()

# Metódo listaSimGrau

## Tabela ECP



## Tabela BVA

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Método | listaSimGrau | | | | |
|  |  |  |  |  |  |
| Cenários |  |  |  |  |  |
| 1 | listaRanking > 0, grau >= 0 | | | | |
| 2 | listaRanking =< 0, grau >= 0 | | | | |
| 3 | listaRanking > 0, grau < 0 | | | | |
|  |  |  |  |  |  |
| Casos de Teste | Input | |  |  | Output |
|  | **listaRanking** | **grau** |  |  |  |
| 1 | 1 | 1 |  |  | lista com similaridade acima do grau |
| 2 | 0 | 1 |  |  | -1 |
| 3 | 1 | 1 |  |  | lista com similaridade acima do grau |
| 4 | 1 | 0 |  |  | lista com similaridade acima do grau |
| 5 | 1 | -1 |  |  | -1 |

## Caso de teste

## Caso de teste 1 ()

## Caso de teste 2 ()

## Caso de teste 3 ()

## Caso de teste 4 ()

## Caso de teste 5 ()

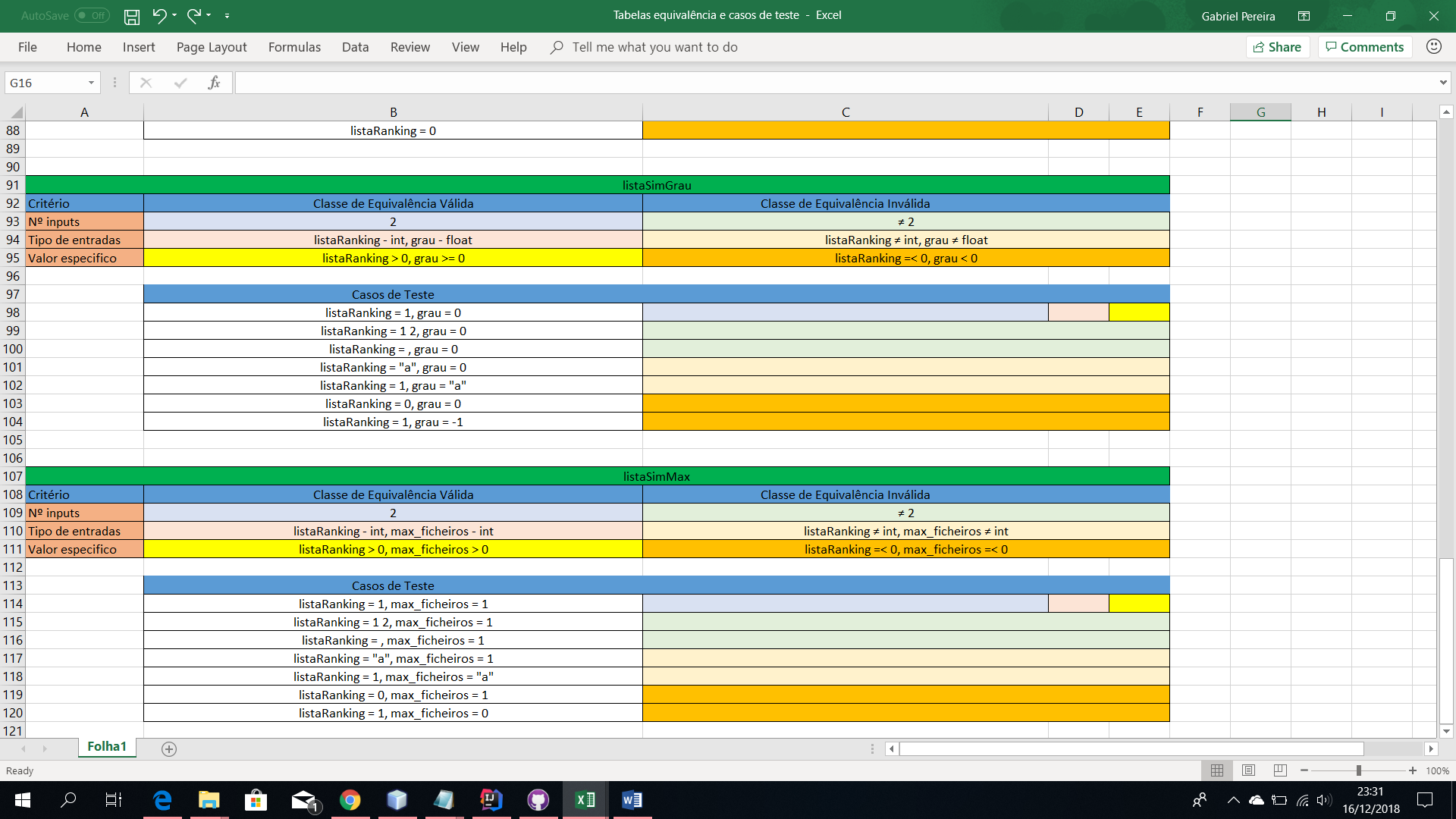
## Caso de teste 6 ()

## Caso de teste 7 ()

## Caso de teste 8 ()

# Metódo listaSimMax

## Tabela ECP



## Tabela BVA

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Método | listaSimMax | | | | |
|  |  |  |  |  |  |
| Cenários |  |  |  |  |  |
| 1 | listaRanking > 0, max\_ficheiros > 0 | | | | |
| 2 | listaRanking =< 0, max\_ficheiros > 0 | | | | |
| 3 | listaRanking > 0, max\_ficheiros =< 0 | | | | |
|  |  |  |  |  |  |
| Casos de Teste | Input | |  |  | Output |
|  | **listaRanking** | **max\_ficheiros** |  |  |  |
| 1 | 1 | 1 |  |  | lista com maximo de ficheiros |
| 2 | 0 | 1 |  |  | -1 |
| 3 | 1 | 1 |  |  | lista com maximo de ficheiros |
| 4 | 1 | 0 |  |  | -1 |

## Caso de teste

## Caso de teste 1 ()

## Caso de teste 2 ()

## Caso de teste 3 ()

## Caso de teste 4 ()

## Caso de teste 5 ()

## Caso de teste 6 ()

## Caso de teste 7 ()

## Caso de teste 8 ()

# Conclusão

Este projeto vem dar resposta a uma solicitação referente ao sprint #1 da Unidade Curricular de Engenharia de Software II que irá ser realizado no dia 17/12/18, para tal usamos as técnicas *Equivalence Class Partitioning* e *Boundary Value Analysis,* descrição de *test inputs, execution conditions* e *expected outputs,* assegurando que os casos de teste cobrem *Valid equivalence classes* e *Invalid equivalence classes.*

Realizamos ainda um estudo das ferramentas que irão ser utilizadas posteriormente no decorrer deste projecto.