|  |  |
| --- | --- |
| LogoPoly_Genie1Classe.png | École Polytechnique de Montréal  Département de Génie Informatique et Génie Logiciel |

INF2990

Projet de logiciel graphique interactif

**Tests logiciels**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Remis par :**

|  |  |
| --- | --- |
| **Matricule** | **Prénom & Nom** |
| **1733229** | **Olivier St-Amour** |
| **1736842** | **Philippe Marcotte** |
| **1739351** | **Simon-Pierre Desjardins** |
| **1740216** | **Ulric Villeneuve** |
| **1744468** | **Frédéric Grégoire** |
| **1745728** | **Camille Gendreau** |

**Le 21 mars 2016**

**Introduction**

**Présentation des travaux**

Pour chacune des classes que vous avez décidé de tester, vous devez :

* identifier la classe testée (et le nom de la classe contenant la suite de cas de test);
* identifier la branche appropriée;
* justifier la pertinence de tester cette classe.

Pour chacune des méthodes que vous avez décidé de tester, vous devez :

* identifier la méthode testée (et le nom de la méthode du cas de test);
* justifier la pertinence de tester cette méthode;
* expliquer ce qu’effectue le cas de test.

Vous pouvez utiliser les tableaux suivant pour entrer les informations.

|  |  |  |  |
| --- | --- | --- | --- |
| **Suite de cas de test #** | | | |
| **Classe testée** | RectangleEnglobant (PhysiqueTest) | **Branche** |  |
|  | | | |
| **Justification** | | | |
|  | | | |

|  |  |
| --- | --- |
| **Cas de test #** | |
| **Méthode testée** | Bool RectangleEnglobant::calculerEstDansForme( const glm ::dvec3& point) const |
|  | |
| **Justification** | |
|  | |
|  | |
| **Explication du cas de test** | |
|  | |
|  | |

|  |  |
| --- | --- |
| **Cas de test #** | |
| **Méthode testée** | Bool RectangleEnglobant::calculerIntersection(const RectangleEnglobant& rectangle) const |
|  | |
| **Justification** | |
|  | |
|  | |
| **Explication du cas de test** | |
|  | |

|  |  |
| --- | --- |
| **Cas de test #** | |
| **Méthode testée** | Bool RectangleEnglobant::calculerIntersection(const CercleEnglobant& cercle) const |
|  | |
| **Justification** | |
|  | |
|  | |
| **Explication du cas de test** | |
|  | |

|  |  |
| --- | --- |
| **Cas de test #** | |
| **Méthode testée** | Bool RectangleEnglobant::assignerAngle(const double& angle) |
|  | |
| **Justification** | |
|  | |
|  | |
| **Explication du cas de test** | |
|  | |
|  | |

|  |  |
| --- | --- |
| **Cas de test #** | |
| **Méthode testée** | Glm::dvec3 RectangleEnglobant::calculerNormaleCollision(const RectangleEnglobant& rectangle) const |
|  | |
| **Justification** | |
|  | |
|  | |
| **Explication du cas de test** | |
|  | |

|  |  |
| --- | --- |
| **Cas de test #** | |
| **Méthode testée** | Glm::dvec3 RectangleEnglobant::calculerNormaleCollision(const CercleEnglobant& rectangle) const |
|  | |
| **Justification** | |
|  | |
|  | |
| **Explication du cas de test** | |
|  | |

|  |  |
| --- | --- |
| **Cas de test #** | |
| **Méthode testée** | Glm::dvec3 RectangleEnglobant::calculerNormaleCollision(const glm::dvec3& point) const |
|  | |
| **Justification** | |
|  | |
|  | |
| **Explication du cas de test** | |
|  | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Suite de cas de test #** | | | |
| **Classe testée** | CercleEnglobant (PhysiqueTest) | **Branche** |  |
|  | | | |
| **Justification** | | | |
|  | | | |

|  |  |
| --- | --- |
| **Cas de test #** | |
| **Méthode testée** | Bool CercleEnglobant::calculerEstDansForme( const glm ::dvec3& point) const |
|  | |
| **Justification** | |
|  | |
|  | |
| **Explication du cas de test** | |
|  | |

|  |  |
| --- | --- |
| **Cas de test #** | |
| **Méthode testée** | Bool RectangleEnglobant::calculerIntersection(const CercleEnglobannt& cercle) const |
|  | |
| **Justification** | |
|  | |
|  | |
| **Explication du cas de test** | |
|  | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Suite de cas de test #** | | | |
| **Classe testée** | FormeEnglobanteAbstraite (PhysiqueTest) | **Branche** |  |
|  | | | |
| **Justification** | | | |
|  | | | |

|  |  |
| --- | --- |
| **Cas de test #** | |
| **Méthode testée** | Void FormeEnglobante::AssignerPositionCentre(const glm ::dvec3& positionCentre) |
|  | |
| **Justification** | |
|  | |
|  | |
| **Explication du cas de test** | |
|  | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Suite de cas de test #** | | | |
| **Classe testée** | SuiveurLigne (CapteurTest) | **Branche** |  |
|  | | | |
| **Justification** | | | |
|  | | | |

|  |  |
| --- | --- |
| **Cas de test #** | |
| **Méthode testée** | Void SuiveurLigne::verifierDetection(NoeudLigne\* ligne) |
|  | |
| **Justification** | |
|  | |
|  | |
| **Explication du cas de test** | |
|  | |

|  |  |
| --- | --- |
| **Cas de test #** | |
| **Méthode testée** | Void SuiveurLigne::obtenirEtatCapteurs() const |
|  | |
| **Justification** | |
|  | |
|  | |
| **Explication du cas de test** | |
|  | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Suite de cas de test #** | | | |
| **Classe testée** | ControleRobot (ControleRobotTest) | **Branche** |  |
|  | | | |
| **Justification** | | | |
|  | | | |

|  |  |
| --- | --- |
| **Cas de test #** | |
| **Méthode testée** | Void SuiveurLigne::obtenirEtatCapteurs() const |
|  | |
| **Justification** | |
|  | |
|  | |
| **Explication du cas de test** | |
|  | |

# Grille de correction

L’évaluation des tests logiciels vaut 5% du cours. Le correcteur appliquera la grille de correction détaillée ci-dessous. Cette grille tient compte des deux éléments suivants :

* la **pertinence** et la **justification** des cas de test choisis parmi les différentes suites de cas de test, ainsi que les tableaux remplis;
* la **qualité** de l’implémentation des cas de test.

## Choix des cas de test

|  |  |
| --- | --- |
| **2,5 pts** | Les cas de test choisis sont **très pertinents**.  **ET**  La justification est **très claire et complète**. |
| **2 pts** | Les cas de test choisis sont **pertinents** ou **très pertinents**.  **ET**  La justification est **généralement claire** et **complète**. |
| **1,5 pt** | Les cas de test choisis sont **pertinents**.  **ET**  La justification est **moyennement claire** et/ou **incomplète**. |
| **1 pt** | Les cas de test choisis sont **peu pertinents**.  **ET/OU**  La justification est **floue** ou **incomplète**. |
| **0 pt** | Les cas de test choisis ne sont **pas pertinents** ou sont **manquants**.  **ET/OU**  La justification est **très floue**, **incomplète** ou **absente**. |

## Implémentation des cas de test

|  |  |
| --- | --- |
| **2,5 pts** | La qualité de l’implémentation des cas de test est **excellente**. |
| **2 pts** | La qualité de l’implémentation des cas de test est **très bonne**. |
| **1,5 pt** | La qualité de l’implémentation des cas de test est **bonne**. |
| **1 pt** | La qualité de l'implémentation des cas de test est **médiocre**. |
| **0 pt** | La qualité de l'implémentation des cas de test est **nettement insuffisante**. |

**Conclusion**