

LOG6305 - TECHNIQUES AVANCÉES DE TEST DU LOGICIEL

LABORATOIRE 2

Département de génie informatique et de génie logiciel
École Polytechnique de Montréal



Hiver 2021

Travail demandé

But : L'application de l'approche SBST (Seach-Based Software Testing) proposée par Wegener (LOG6305 sbse-testing-09-EN, Page 95-96) au problème de génération des trois côtés du triangle.

Partie 1 - Préparation des données

1. **Conditions uniques :** Énumérez et nommez toutes les conditions uniques dans la classe *triangle.py*.
2. **Chemins d'exécution :** Représentez chaque chemin d'exécution par une séquence des conditions uniques sous la forme : $\text{Chemin}_i : \{cond_1, cond_2, \dots, cond_n\}$.
3. **Graphe de flot de contrôle :** Dessinez [le graphe de flot de contrôle](#) contenant tous les chemins d'exécution dans la classe *triangle.py*.

Partie 2 - Random Search

1. Générez les données initiales (3 côtés d'un triangle) de façon aléatoire.
2. Comparez les données générées avec un chemin d'exécution, ou chaque nœud du chemin doit être satisfait par les données
3. Ajustez les données et comparez-les à nouveau du nœud supérieur du chemin. Répétez cette étape jusqu'à ce que tous les nœuds d'un chemin soient satisfaits par les données.
4. Répétez les deux étapes 2-3 pour toutes les branches d'exécution de la classe *triangle.py*.

Partie 3 - SBST

1. **SearchBasedTest :** Ajoutez une nouvelle classe *SearchBasedTest* qui hérite de la classe *CoverageTest*
 - (a) Définissez les opérateurs de comparaisons comme $>$, $==$, $<$, *etc.*
 - (b) Représentez les données d'entrée du triangle (par exemple, une variable pour chacun des côtés).
 - (c) Ajoutez une classe/structure de données *ExecPath* pour stocker tous les chemins d'exécution possibles de la classe *triangle* de manière que vous pouvez itérer sur les chemins d'exécutions et leurs nœuds correspondants sur lesquels vous calculez la fitness des données générées.
 - (d) Implémentez une méthode pour sortir la distance des conditions sur le parcours à suivre.
2. **FitnessFunction :** Implémentez une nouvelle classe *fitnessFunction* qui permet de calculer la fitness basé sur l'approche de Wegener.
3. **Génération des données :** Implémentez une nouvelle méthode *"SearchBasedTest.generateTestData"* qui se base sur l'approche SBST (escalade de la colline) qui utilise la *fitnessFunction* définie .
4. **Documentation :** Les chemins d'exécution doivent être clairement représentés dans le code ainsi que chaque condition ou chemin d'exécution doit être clairement commenté.

Livrables attendus

Les livrables suivants sont attendus :

- un rapport **pdf** qui doit contenir le titre et numéro du laboratoire, les noms et matricules des coéquipiers.
- un dossier contenant votre code commenter.

Le tout à remettre dans une seule archive **zip** avec pour nom matricule1_matricule2_lab1.zip à télé-verser sur Moodle.

Consultez le site Moodle du cours pour la date et l'heure limites de remise des fichiers. Un retard de]0,24h] sera pénalisé de 10%, de]24h, 48h] de 20% et de plus de 48h de 50%.