

# TP1 : Fiabilité logicielle

10 janvier 2023

ARTAUD Cyril  
DRAOU Meryem

Nous regardons dans un premier temps la classe `TriangleTest`.

On lance d'abord les tests `testGetterEdgeLengthA()`, `testGetterEdgeLengthB()`, `testGetterEdgeLengthC()`, puis on constate que les tests passent.

Ensuite on lance tous les tests `testSetEdgeLengthA()`, `testSetEdgeLengthB()`, `testSetEdgeLengthC()` et on constate que les trois tests passent.

Nous vérifions ensuite avec les tests `testTypeEquilateral()`, `testTypeIsocèles()`, `testTypeScalene()`, la méthode `TriangleType type()` et on constate que nos tests passent quel que soit le type de triangle, notamment à l'aide de la méthode `assertEquals()` utilisée pour tester l'égalité entre deux valeurs.

Ensuite on test également si notre fichier est correct avec la méthode `testTriangleFile()`.

Nous regardons dans un second temps la classe `search_ArrayTest`.

Pour la recherche dans un tableau trié, nous vérifions ensuite avec la méthode `initArray()` et avec le mot clé `@BeforeAll` que notre tableau est correct pour la suite des test.

Nous vérifions la méthode `search()` de chaque classe, avec les méthodes `search_Array_Class_1_Test()`, `search_Array_Class_2_Test()`, `search_Array_Class_3_Test()` et `search_Array_Class_4_Test()`.

Pour le premier test, nous recherchons l'élément 0 que nous trouvons dans le tableau trié (le test passe).

Pour le second test, nous recherchons l'élément 1 que nous trouvons dans le tableau trié (le test passe).

Pour le troisième test, nous recherchons l'élément 2 (le test ne passe pas et boucle à l'infini, comme souhaité).

Pour le quatrième test, nous recherchons l'élément 3 que nous trouvons dans le tableau trié (le test passe).