

DÉPARTEMENT DE MATHÉMATIQUES, D'INFORMATIQUE ET DE GÉNIE

**INF33307 – Assurance de la qualité et gestion**

**de projets informatiques**

**Automne 2021**

**Solution du travail pratique #3 (partie A)**

**Professeur: Ismaïl Khriss**

**Auxiliaire d’enseignement : Mamadou Mbow**

# 2.1 Complexité cyclomatique

Une image contenant intérieur

Description générée automatiquement

CC = E – N + 2 = 11 – 9 + 2 = 4

# 2.2 Chemins linéairement indépendants

Chemin 1 : 1-7, 8, 26

Chemin 2 : 1-7, 8, 9-11, 12, 25, 8, 26

Chemin 3 : 1-7, 8, 9-11, 12, 13-15, 24, 12, 25, 8, 26

Chemin 4 : 1-7, 8, 9-11, 12, 13-15, 16-23, 24, 12, 25, 8, 26

# 2.3 Tests unitaires .NET

**Chemin 1 : impossible à tester**

Une image contenant texte

Description générée automatiquement

Une image contenant texte

Description générée automatiquement

# 2.4 Erreurs et tests de régression

**Échec des tests 3, 4 :**

- La boucle qui parcourt le tableau commence à l’index 1 au lieu de 0 (ligne 12), ce qui signifie que le premier élément ne sera pas pris en considération lors du tri.

- Dans la même ligne, il faut modifier la condition j < n par j < (n - 1) car sinon l’algorithme essaierai de comparer le dernier élément du tableau avec un élément ayant un index dépassant la taille du tableau au niveau de la ligne 15.

**Version corrigée de l’algorithme**

void tri\_a\_bulle(int[] t, int n)

{

int tmp = 0; // Variable de stockage temporaire

// Booléen marquant l'arrêt du tri si le tableau est ordonné

bool en\_desordre = true;

/\* Boucle de répétition du tri et le test qui

arrête le tri dès que le tableau est ordonné (en\_desordre=false) \*/

while(en\_desordre)

{

// Supposons le tableau ordonné

en\_desordre = false;

for(int j = 0 ; j < (n – 1) ; j++)

{

// Si les 2 éléments sont mal triés

if (t[j] > t[j+1])

{

/\* Inversion des 2 éléments \*/

tmp = t[j+1];

t[j+1] = t[j];

t[j] = tmp;

// Le tableau n'est toujours pas trié

en\_desordre = true;

}

}

}

}

**Tests de régression :**

# Une image contenant texte Description générée automatiquement

# 2.5 Discussion sur l’exactitude de la méthode

Oui, il est possible que la méthode soit encore incorrecte même si tous les tests de régression ont réussi.

La complexité cyclomatique permet de répertorier les chemins importants à partir desquels on peut concevoir des cas de test. Cependant, elle ne permet pas de concevoir tous les cas possibles de l’algorithme. En effet, afin de pouvoir affirmer que la méthode soit fonctionnelle à 100%, il faudrait être en mesure de prouver qu’elle fonctionne correctement dans tous les cas. Ce qui revient à faire des tests exhaustifs. Ce qui est en pratique impossible. D’autres techniques de test sont possibles pour concevoir d’autres cas de tests pour augmenter la couverture de notre suite de tests.