

TD série n°2: techniques de test structurel

L'objectif de cette deuxième série est de mettre en œuvre dans la pratique certaines techniques de **test structurel** introduites en cours, en abordant les aspects :

- **statique**: pas d'exécution de programme
- et **dynamique**: exécution de programme sur un jeu de DT généré selon la technique de couverture.

Exercice 1: Le pseudo code ci-dessous est censé représenter un composant qui cherche l'indice d'un élément (s'il existe) dans une liste triée.

```
look (ELEMENT cle, ELEMENT tab[ ], integer taille, boolean trouve, integer A)
begin
  integer droit, gauche, median, inf, sup ;
  boolean inc;
  gauche :=inf; droit :=sup;
  A :=(droit + gauche)/ 2 ;
  trouve:= inc;
  if tab[A] = cle
    then  trouve :=true;
    else  trouve :=false;
  while gauche <= droit and trouve = false
    begin
      median:=(droit + gauche) / 2;
      if tab[median] = cle
        then
          begin
            trouve:=true;
            A:=median
          end
        else
          if tab[median]> cle
            then gauche:=median+1;
            else droit :=median-1;
          end
        end
    end
end look
```

- 1- Décrire le **graphe de contrôle** correspondant.
- 2- Calculer le nombre **cyclomatique** et donner sa signification pour le code.
- 3- Etablir l'**expression des chemins** de contrôle.
- 4- Etablir le tableau des $\pi()$ expressions(ou **dr-chaînes**) correspondant aux différentes variables.
- 5-Dresser le tableau qui met en évidence d'éventuelles anomalies dans le flot de données.
- 6- En utilisant les résultats qui précèdent, proposer une **mise au point** de la procédure sous la forme d'un code C, C++ ou Java pour obtenir le composant logiciel sous test.
- 7-Générer un jeu de DT pour sensibiliser les chemins exécutables permettant la couverture de **tous les nœuds** du graphe de contrôle, déterminer le taux de couverture de chaque DT.
- 8-Générer un jeu de DT pour sensibiliser les chemins exécutables permettant la couverture de **tous les arcs** du graphe de contrôle, déterminer le taux de couverture de chaque DT.
- 9-Générer un jeu de DT pour sensibiliser **tous les chemins indépendants** du graphe et déterminer le taux de couverture de chaque DT.
- 10-Exécuter le composant logiciel sous test sur les données générés en 7), 8) et 9.

Exercice 2: Reconnaissance d'une signature radar

Le radar d'un avion de combat détecte les signatures émises par les différents types de missiles qui, potentiellement, peuvent constituer une menace. Il reconnaît toutes les **séquences** de fréquences(ou signatures) des signaux détectés qui :

- commencent par les fréquences **F00 ou F01**
- suivies d'une sous-suite σ composée de fréquences **F10 ou F11 ou F12**
- et se terminent par les fréquences **F20 ou F21**

Le nombre d'apparitions de chaque fréquence dans σ est enregistré dans un compteur: **count0** pour **F10**, **count1** pour **F11** et **count2** pour **F12**.

Dans l'hypothèse où la menace se limite à trois types de missiles : A, B et C, l'identification est réalisée en appliquant l'«algorithme approché» suivant :

- si $\text{count0} < N0$ ou $\text{count1} < N1$ ou $\text{count2} < N2$ (où $N0$, $N1$ et $N2$ appelés «paramètres de fiabilité» du système sont connus et tels que $N0 > N1 > N2$), la signature ne correspond à aucun des trois types de missiles.
- si $\log(\text{count0}) \geq \log(\text{count1} + 4 \text{count2})$ alors le missile détecté est de type A,
- si $\log(\text{count1}) \geq \log(2 \text{count2})$ alors le missile détecté est de type B,
- toute autre configuration correspond au missile de type C.

0-Proposer un code source (c, c++ ou java) du composant logiciel sous test

1. Représenter ce code à l'aide d'un **graphe de contrôle**

2. Calculer le **nombre cyclomatique** et en déduire le nombre de décisions simples.

3. Etablir l'**expression des chemins** de contrôle.

4-Générer un jeu de DT pour sensibiliser les chemins exécutables permettant la couverture de **tous les noeuds** du graphe de contrôle, déterminer le taux de couverture de chaque DT.

5-Générer un jeu de DT pour sensibiliser les chemins exécutables permettant la couverture de **tous les arcs** du graphe de contrôle, déterminer le taux de couverture de chaque DT.

6-Générer un jeu de DT pour sensibiliser **tous les chemins indépendants** du graphe et déterminer le taux de couverture de chaque DT.

7-Exécuter le programme sur les données générés en 4), 5) et 6.