

	ING2 – Computer Systems Engineering Test et Vérification - TD2	
	Subject : Informatique	Date : Novembre 2020
		Duration :
		Number of pages : red2page.2

1 Tri bicolore

Exercise 1.

Soit le code suivant :

```

1 public static int[] triBicolore(int[] t) {
2     int i; // start iterator
3     int izero; // last index for next zero (last box)
4     i = 1;
5     izero = t.length - 1;
6     while (i < izero) {
7         // if read a zero, swap with last index
8         if (t[i] == 0) {
9             t[i] = t[izero];
10            t[izero] = 0;
11            izero--;
12        }
13        i++;
14    }
15    return t;
16 }
```

Figure 1: Code 1

Ce code réalise le tri suivant : pour un tableau composé aléatoirement de 0 et de 1, il place les 1 de manière contigüe, au début et les zéros à la fin.

a. Écrire des cas de test qui :

- n'exécute pas la boucle while,
- exécute **exactement une fois** la boucle while.

b. Ajouter des cas de test exhaustifs pour valider (ou non) la méthode (corriger le code si nécessaire).

2 Caractère le plus présent

Exercise 2.

Écrire des cas de test qui vérifient la correction du code qui trouve le caractère le plus présent dans une chaîne (Figure 2).

```

1 private static char countMaxCar( String input) {
2     Map<Character , Integer> charMap = new HashMap<>();
3     int maxi = 1;
4     char res;
5     for ( char key : input.toCharArray() ) {
6         if ( charMap.containsKey(key) ) {
7             cpt = charMap.get(key);
8             charMap.replace(key, cpt++);
9         }
10        charMap.put(key, 1);
11    }
12    for ( char key : charMap.keys() ) {
13        if (charMap.get(key) > maxi)
14            res = key;
15    }
16    return res;
17 }

```

Figure 2: Code 2

3 Triangles

Exercice 3. L'énoncé du problème est le suivant : Ecrire un programme qui lit 3 entiers depuis une zone de saisie. Les 3 valeurs représentent respectivement les longueurs des côtés d'un triangle. Le programme doit retourner un message qui détermine si le triangle est scalène, isocèle ou équilatéral.

Pour simplifier, on va créer une classe Java contenant simplement un constructeur Triangle(int,int,int) et 3 méthodes public: boolean scalene(), boolean isocèle() et boolean equilateral().

Nous ne traiterons donc pas la saisie dans un premier temps

- a. Écrire une classe Java pour ce programme.
- b. Écrire une classe de tests.
- c. Évaluez la qualité de vos tests en répondant aux 14 questions posées par votre professeur.
- d. Ajoutez des tests et corrigez votre programme en fonction des résultats de la question précédente.
- e. Vous devriez également passer tous les tests de la classe TriangleTest en pièce jointe. Si ce n'est pas le cas, modifiez votre programme et vos tests.