

# Exercice 1 – Révision

22 janvier 2024

**Préparé par**

Benjamin Lemelin, François Paradis, Pierre-François Léon et Karine Filiatreault

# 1 Travail à effectuer

Commencez par créer un nouveau projet C# de type « Application Console ».

Ajoutez le mot clé « public » devant la class Program.

```
public class Program
{
```

## 1.1 Indice de masse corporelle

Créez une fonction calculant [l'indice de masse corporelle](#) d'un individu, dont voici la formule :

$$IMC = \frac{p}{t^2}$$

Où :

- $p$  est le poids en kilogrammes.
- $t$  est la taille en mètres.

Par exemple, pour un individu de **58,9 kg** mesurant **1,72 m**, l'indice de masse corporelle est de **19,9**. Essayez quelques fois votre nouvelle fonction dans la fonction « **main** ».

Lorsque votre fonction est écrite, appuyez sur le bouton droit de la souris et choisir « Créer des tests unitaires ». Faites « OK » dans la fenêtre suivante.

Créez des tests unitaires et exécutez-les. Assurez-vous de couvrir tous les cas possibles. Nommez chaque fonction de test de façon précise. Voir le fichier d'exemples fourni.

Rappel des fonctions d'assertions les plus courantes – n'ayez pas peur de regarder la documentation :

- Assert.AreEqual ou .AreNotEqual
- Assert.IsTrue ou .IsFalse
- Assert.AreSame ou .NotSame
- CollectionAssert.AreEqual
- CollectionAssert.AreEqual

## 1.2 Valeur minimale et valeur maximale

Soit le tableau suivant :

```
int[] values = {42, 8, 10, -1, 0, 4, 3};
```

Créez une fonction retournant la valeur minimale dans ce tableau, ainsi qu'une autre fonction retournant la valeur maximale.

Créez des tests unitaires et exécutez-les. Assurez-vous de couvrir tous les cas possibles.

### 1.3 Recherche

Écrivez une fonction qui prend en paramètres un tableau d'entiers et une valeur à chercher. Cette fonction renvoie vrai si la valeur est trouvée, faux sinon.

Créez des tests unitaires et exécutez-les. Assurez-vous de couvrir tous les cas possibles.

### 1.4 Vérifier tri

Écrivez une fonction qui prend en paramètres un tableau d'entiers. Cette fonction renvoie vrai si le tableau est trié en ordre croissant, faux sinon.

Créez des tests unitaires et exécutez-les. Assurez-vous de couvrir tous les cas possibles.

### 1.5 Inverser

Écrivez une fonction qui prend en paramètres un tableau d'entiers et qui renvoie un nouveau tableau dont les valeurs sont rangées dans l'ordre inverse.  $\{1, 4, 3\} \Rightarrow \{3, 4, 1\}$ .

Créez des tests unitaires et exécutez-les. Assurez-vous de couvrir tous les cas possibles.

### 1.6 Échange

Écrivez une fonction qui prend en paramètres un tableau d'entiers, deux indices et qui échange les valeurs présentes aux indices donnés. La fonction agit sur le tableau passé en paramètres.

Créez des tests unitaires et exécutez-les. Assurez-vous de couvrir tous les cas possibles.

### 1.7 Somme des tableaux

Écrivez une fonction qui prend en paramètres deux tableaux de même capacité et qui renvoie un nouveau tableau dont chaque case est la somme des valeurs correspondantes dans les deux tableaux.

Créez des tests unitaires et exécutez-les. Assurez-vous de couvrir tous les cas possibles.

## 1.8 Insérer

Écrivez une fonction qui prend en paramètres un tableau de réels, un indice et une nouvelle valeur à insérer dans le tableau. La fonction doit insérer la nouvelle valeur à l'indice reçu en paramètre, en décalant les éléments existants. Le dernier élément sera perdu. La fonction agit sur le tableau passé en paramètres.

Par exemple :

- Avec le tableau « `int tab[6] = {15, 25, -3, 6, 27, 13};` »
- Après avoir exécuter « `Inserer(tab, 2, -99);` » le tableau devrait contenir : {15, 25, -99, -3, 6, 27}

Créez des tests unitaires et exécutez-les. Assurez-vous de couvrir tous les cas possibles.

## 2 Modalités de remise

Remettez votre projet C# sur LÉA, dans la section Travaux, à l'intérieur d'une archive *Zip*. Supprimez tous les fichiers et dossiers temporaires (bin, obj, TestResults et .vs).