

## EXERCICE 08 – FONCTIONS 1

Préparé par Benjamin Lemelin et Pierre Poulin le 18 janvier 2023

## 1 Travail à effectuer

Ce travail consiste à reprendre du code des exercices précédent et à les transformer en fonctions prenant une entrée et produisant une sortie. Par exemple, cet extrait de code ...

```
public static void FindSmallerNumber2Values()
{
```

```
    //Entrées
```

```
    int number1 = 50;
```

```
    int number2 = 10;
```

```
    // Algorithme
```

```
    int smallest;
```

```
    if (number1 <= number2)
```

```
    {
```

```
        smallest = number1;
```

```
    }
```

```
    else
```

```
    {
```

```
        smallest = number2;
```

```
    }
```

```
    // Sortie
```

```
    // Valider la valeur de la variable smallest
```

```
}
```

... devient ...

```
public static int FindSmallerNumber2Values(int number1, int number2)
```

```
{
```

```
    int smallest;
```

```
    if (number1 <= number2)
```

```
    {
```

```
        smallest = number1;
```

```
    }
```

```
    else
```

```
    {
```

```
        smallest = number2;
```

```
    }
```

```
    return smallest;
```

```
}
```

Cette fois, le projet de départ qui vous est donné avec cet exercice ne contient presque rien et il vous faudra donc écrire la majorité du code vous-même.

**Prudence**

Il va de soi que si votre code ne fonctionnait pas, vous allez devoir le corriger avant de commencer cet exercice.

## 1.1 Frais de scolarité

Localisez la fonction **ComputeTuitionFees** de l'exercice 05 et copiez la dans cet exercice. Transformez cette fonction tel qu'indiqué à la page précédente. Vous devriez donc avoir une fonction prenant en paramètre un statut de résidence (un booléen), un nombre de cours (entier) et retournant le coût total des frais de scolarité (un nombre à virgule).

Dans votre **Main**, appelez cette fonction pour chacun de ces tests et vérifiez la sortie dans le débogueur.

#	Variables	
	Entrées	Sortie
1	isResident = true nbCourses = 5	result = 750.00
2	isResident = false nbCourses = 5	result = 1600.00
3	isResident = true nbCourses = 0	result = 0.00

## 1.2 Prix d'un ticket

Localisez la fonction **ComputeTicketPrice** de l'exercice 05 et copiez la dans cet exercice. Transformez cette fonction tel que vous l'avez fait au numéro précédent. Vérifiez que votre fonction marche en l'appelant dans le **Main** avec ces valeurs :

#	Variables	
	Entrées	Sortie
1	age = 11 isSkating = false isSkiing= false	result = 2.00
2	age = 11 isSkating = true isSkiing= false	result = 4.00
3	age = 11 isSkating = true isSkiing= true	result = 7.00
4	age = 12 isSkating = false isSkiing= false	result = 4.00
5	age = 12 isSkating = true isSkiing= false	result = 8.00
6	age = 12 isSkating = true isSkiing= true	result = 14.00
7	age = 13 isSkating = false isSkiing= false	result = 4.00
8	age = 13 isSkating = true isSkiing= false	result = 8.00
9	age = 13 isSkating = true isSkiing= true	result = 14.00

### 1.3 Sommes de chiffres qui composent un nombre

Localisez la fonction `ComputeSumOfDigits` de l'exercice 06 et copiez la dans cet exercice. Transformez votre code en une fonction tel que vous l'avez déjà fait et vérifiez qu'elle marche en l'appelant dans le `Main` avec ces valeurs :

#	Variables	
	Entrées	Sortie
1	<code>number = 0</code>	<code>sumOfDigits = 0</code>
2	<code>number = 2</code>	<code>sumOfDigits = 2</code>
3	<code>number = 15</code>	<code>sumOfDigits = 6</code>
4	<code>number = 48</code>	<code>sumOfDigits = 12</code>

### 1.4 Somme des nombres pairs

Pour ce dernier numéro, vous allez devoir écrire une toute nouvelle fonction. Cette fonction devra calculer la somme des  $N$  premiers nombres pairs (en commençant par 0). Par exemple, la somme des 5 premiers nombres pairs est 20, c'est-à-dire  $0 + 2 + 4 + 6 + 8 = 20$ .



#### Prudence

Nous considérons le nombre 0 comme étant le premier nombre pair.

Comme toujours, appelez cette fonction dans votre `Main` avec ces valeurs :

#	Variables		
	Entrées	Calcul	Sortie
1	<code>nbEvenNumbers = 0</code>	--	0
2	<code>nbEvenNumbers = 1</code>	0	0
3	<code>nbEvenNumbers = 2</code>	$0 + 2$	2
4	<code>nbEvenNumbers = 5</code>	$0 + 2 + 4 + 6 + 8$	20



#### Information

Vous aurez probablement remarqué que nous ne vous avons pas indiqué comment nommer votre fonction. C'est voulu. Le nommage est une compétence qui s'acquiert avec le temps, et il faut le pratiquer le plus possible.

## 2 Modalités de remise

Remettez votre projet Visual Studio sur LÉA, dans la section travaux, à l'intérieur d'une archive *Zip*. Supprimez tous les dossiers temporaires, à savoir les dossiers `.vs`, `TestResults`, `bin` et `obj`.