

# Travaux Pratiques 1

## Objectifs:

- Déclarer des variables
- Utiliser les opérateurs (+, %)
- Utiliser les instructions de sélection (if, else)
- Utiliser les instructions d'itération (for, while)
- Manipuler les nombres entiers et les chaînes
- Comprendre les palindromes et les nombres premiers
- Afficher un résultat dans la console

## Critères de performances:

- 3.1 Résolution efficace des problèmes de conception de l'algorithme.
- 3.2 Résolution efficace des problèmes de traduction de l'algorithme dans le langage de programmation.
- 3.3 Résolution efficace des problèmes d'utilisation du langage de programmation.
- 4.3 Interprétation juste des résultats.
- 4.4 Fonctionnement correct du programme.

Date de remise: 7 février 2018

## Instructions:

1. Téléchargez le projet Visual Studio ici:  
<https://github.com/felixsoul/420JV4AS-TP1>
2. Écrivez votre nom en commentaire au début du fichier Program.cs
3. Faites tout le travail dans Program.cs et soumettez seulement ce fichier sur Léo

Chaque question correspond à une fonction. Les fonctions doivent être implémentées correctement pour que l'appel de ces fonctions affiche des résultats qui correspondent à ceux dans ce document.

## Question 1. Fizz Buzz

Fizz Buzz est une question simple avec plusieurs implémentations possibles, souvent utilisée dans les entrevues. Affichez les chiffres de 1 à 100, mais en suivant les conditions suivante:

- Si le chiffre est divisible par 3, remplacez le par “Fizz”
- Si le chiffre est divisible par 5, remplacez le par “Buzz”
- Si le chiffre est divisible par 3 et 5, remplacez le par “FizzBuzz”

Résultats:

```
// Question1()
1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
Buzz
11
Fizz
13
14
FizzBuzz
16
17
Fizz
19
Buzz
...
```

## Question 2. Les chaînes

(a) Implémentez une fonction pour échanger la première moitié d'une chaîne avec l'autre moitié. Si la chaîne a une longueur impaire, la première moitié aura un caractère de moins que l'autre moitié. Affichez le résultat dans la console.

Résultats:

```
// Question2a("HelloWorld")  
WorldHello
```

```
// Question2a("EggsBacon")  
BaconEggs
```

(b) Implémentez une fonction pour compter le nombre de fois qu'un certain caractère se retrouve dans une chaîne. Affichez le résultat dans la console.

Résultats:

```
// Question2b("HelloWorld", 'a')  
0
```

```
// Question2b("HelloWorld", 'e')  
1
```

```
// Question2b("HelloWorld", 'o')  
2
```

(c) Implémentez une fonction qui analyse une chaîne et qui affiche True si c'est un palindrome, sinon False. Un palindrome est un mot dont l'ordre des lettres restent le même de gauche à droite et de droite à gauche.

Résultats:

```
// Question2c("tacocat")  
True
```

```
// Question2c("nyancat")  
False
```

### Question 3. Les nombres premiers

(a) Le plus grand commun diviseur (PGCD) de deux nombres entiers est le nombre entier le plus grand qui peut diviser les deux. Implémentez une fonction qui affiche le PGCD de deux nombres entiers.

Résultats:

```
// Question3a(4, 20)
4
```

```
// Question3a(6, 8)
2
```

```
// Question3a(7, 8)
1
```

(b) Implémentez une fonction qui analyse un nombre entier et qui affiche True si c'est un nombre premier, sinon False. Un nombre premier c'est un nombre qui a exactement 2 diviseurs distincts: 1 et lui-même.

Résultats:

```
// Question3b(7)
True
```

```
// Question3b(420)
False
```

```
// Question3b(1)
False
```

(c) Implémentez une fonction qui prends comme entré un nombre entier n, puis qui affiche le nième nombre premier.

Résultats:

```
// Question3c(100)
541
```

```
// Question3c(1000)
7919
```

## Question 4. Azulejo.

Un azulejo est un oeuvre d'art composé de plusieurs tuiles en céramiques. Implémentez une fonction pour afficher divers motifs de caractères. À chaque itération dans la boucle, il faut analyser l'indexe pour choisir le bon caractère à afficher. Le paramètre d'entrée de la fonction est un nombre entier qui détermine la longueur en tuiles.

(a) Les diagonales.

Résultats:

```
// Question4a(4)
. @ . @
@ . @ .
. @ . @
@ . @ .
```

```
// Question4a(8)
. @ . @ . @ . @
@ . @ . @ . @ .
. @ . @ . @ . @
@ . @ . @ . @ .
. @ . @ . @ . @
@ . @ . @ . @ .
. @ . @ . @ . @
@ . @ . @ . @ .
```

(b) L'échiquier

Résultats:

```
// Question4b(4)
. . @ @
. . @ @
@ @ . .
@ @ . .
```

```
// Question4b(8)
. . @ @ . . @ @
. . @ @ . . @ @
@ @ . . @ @ . .
@ @ . . @ @ . .
. . @ @ . . @ @
. . @ @ . . @ @
@ @ . . @ @ . .
@ @ . . @ @ . .
```

(c) Les quatres diamants

Résultats:

```
// Question4c(8)
```

A 10x10 grid of dots. The dots are arranged in a regular pattern. Some dots are replaced by the letter 'a'. Specifically, the dots at positions (row, column) where both row and column are even (assuming the top-left dot is at (1,1)) are replaced by 'a'. The 'a's are located at (2,2), (2,4), (2,6), (2,8), (4,2), (4,4), (4,6), (4,8), (6,2), (6,4), (6,6), (6,8), (8,2), (8,4), (8,6), and (8,8).

```
// Question4c(32)
```

A 20x20 grid of dots. The pattern consists of 'D' characters placed at specific grid intersections. The 'D's form a central cross shape, with additional clusters in the four corners. Specifically, 'D's are located at positions where the row and column indices are equal (forming a diagonal), or where the row and column indices are equal to 10 (forming a horizontal and vertical cross). Additionally, there are 'D's at positions where the row and column indices are equal to 19 (forming a diagonal in the bottom-right corner) and at positions where the row and column indices are equal to 1 (forming a diagonal in the top-left corner).