

**Sujet d'examen - 1<sup>ère</sup> session**  
**Année universitaire 2022-2023**

**Intitulé de l'épreuve :** TP Algorithmique

**Nom de l'enseignant :** Mouhammad SAID

Mention / Spécialité / Parcours : Master MSI

Année : 1

Durée de l'épreuve : 2 h

**SUJET** (2 pages)

Lors de démarrage pensez à créer un nouveau projet Java dans IntelliJ appelé **ExamenTPAlgo** contenant une classe **ExamenTP\_VotreNomPrenom**. C'est cette classe qui contiendra votre fonction **public static void main(String[] args)** et les méthodes ci-dessous.

**Exercice 1 (4 points)**

Écrivez une méthode **sommeHarmonique(Integer n)** qui prend en paramètre un entier n et qui calcule la somme des n premiers termes de la "série harmonique", c'est-à-dire la somme :

$$1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{n}$$

**Par exemple :**

Si n = 5, alors **sommeHarmonique(5)** affichera : La somme des 5 premiers termes = 2.2833335.

Si n = 6, alors **sommeHarmonique(6)** affichera : La somme des 6 premiers termes = 2.4500003.

**Exercice 2 (3 points)**

Un entier positif k est un **nombre parfait** s'il est égal à la somme de tous ses diviseurs propres. Par exemple, 6 est parfait car  $6 = 1 + 2 + 3$ . Écrivez une méthode **parfait** qui prend un entier positif k en argument et qui renvoie un booléen indiquant si k est un nombre parfait.

**Par exemple :**

**parfait(6)** renvoie : true

**parfait(15)** renvoie : false

### Exercice 3 (3 points)

Écrivez une méthode **afficheEntiers(Integer n)** qui affiche les entiers jusqu'à n en remplaçant les multiples de 5 par M5, les multiples de 7 par M7 et les multiples des deux par M57.

**Par exemple :**

Si n = 36, alors `afficheEntiers(36)` affichera : 2 3 4 M5 6 M7 8 9 M5 11 12 13 M7 M5 16 17 18 19 M5 M7 22 23 24 M5 26 27 M7 29 M5 31 32 33 34 M57 36.

### Exercice 4 (4 points)

Écrivez une méthode **genererMot(char[] tab1, Integer[] tab2)** qui prend en argument un tableau de caractères tab1 et un tableau d'entiers tab2 et qui affiche le mot obtenu en concaténant les lettres du tableau tab1, comme suit : la lettre sur la position i dans tab1 est répétée autant de fois que l'indique le numéro sur la position i dans tab2. La méthode doit afficher « Erreur » si les deux tableaux n'ont pas la même longueur.

**Par exemple :**

```
char[] tab1 = {'a','b','c','d'}  
Integer[] tab2 = {2,2,3,4}  
genererMot(tab1,tab2) affichera "aabbcccdddd".
```

### Exercice 5 (6 points)

Écrivez une méthode **difference(Integer[] tab1, Integer[] tab2)** qui prend en argument deux tableaux d'entiers tab1, et tab2 (considérés sans doublons) et qui renvoie le tableau représentant la différence symétrique de tab1 et tab2.

NB : La « différence symétrique » de A et B est l'ensemble des éléments appartenant à A ou à B exclusivement.

**Par exemple :**

```
Integer[] tab1 = {6,20,12,1000,8}, tab2 = {2,8,6,7,12}  
difference(tab1,tab2) renvoie {20,1000,2,7}
```

**Bon Courage**