



Colle S1

Anagram Master

Tiens, tu as vu ? **Image** c'est l'anagramme parfait de **magie** ! Oh, et **Marie** celui d'**aimer** ! Trop fort. Moi aussi j'aimerais devenir le maître incontesté des anagrammes ... Cela donnerait un peu plus de sens à ma vie. Aller, je m'y mets, je vais me coder un petit outil en **PHP** qui va me permettre de connaître tous les anagrammes possibles et imaginable dans la langue Française.



Coïncidence, je ne crois pas

Description du document

Titre	Anagram Master
Date	05/09/2016
Auteur	Sébastien Roth
Responsable	Sophie Viger
E-mail	pedagowac@epitech.eu
Sujet	Créer un programme qui donne les anagrammes d'un mot donné
Mots-clés	Anagrammes
Version	1.0

Tableau des révisions

Date	Auteur	Sections	Commentaire
05/09/2016	Sebastien Roth	Toutes	Création du document.
14/09/2016	Sébastien Roth	Toutes	Correctifs + ajouts d'exemples

Table of Contents

Introduction	6
Le programme en lui-même	6
Quelques exemples.....	7

Introduction

Bonjour à tous ! L'objectif de la présente colle est de réaliser dans le temps imparti un outil qui permet de générer des anagrammes par rapport à un mot donné, qui se situe dans le dictionnaire de la langue française.

Pour tous ceux qui auraient oublié malencontreusement ce qu'est une anagramme vous pouvez bien sûr vous rendre sur votre site favoris ... j'ai nommé [Wikipédia](https://fr.wikipedia.org/) !



Procédure

Votre programme devra générer, à partir d'un mot rentré en paramètre, la liste de ses anagrammes parfaites. Si un deuxième argument (nb) est donné, vous listerez l'ensemble de ses anagrammes avec un nombre nb de lettres qui ne correspondent pas au mot donné. Toutefois, ces mêmes anagrammes « imparfait » auront le même nombre de lettre que le mot rentré en paramètre.

Vous **devez** travailler avec le **dictionnaire fourni**. C'est à dire que si vous rentrez un mot dans le programme, il se doit d'être dans le dictionnaire, sinon ça ne marche pas.

Un même mot ne peut pas être sa propre anagramme.

Le nom du dépôt de rendu est : **anagram-master**

Le nom du programme est : **anagram.php**

N'oubliez pas de donner les bons droits de ramassage au rendu.



Obligation : Votre script devra être écrit en PHP. Faire le programme dans un autre langage entrainera un -42.

Obligation 2 : Votre programme devra se lancer en ligne de commande dans votre terminal. Sinon vous ne serez pas noté.

Obligation 3 : Vous devez faire un seul et même fichier. Nous prendrons la peine de n'en corriger qu'un seul dans le cas où vous rendez plusieurs.



Des fonctions ont été interdites. Veuillez-vous référer à la partie restriction située à la fin de l'énoncé

Introduction

Vous devez faire un fichier nommé **anagram.php**.

Vous devez obligatoirement respecter ce format là pour lancer le programme :

USAGE

`php anagram.php s1 (nb)`

DESCRIPTION

<code>s1</code>	Obligatoire	le mot recherché
<code>nb</code>	Optionnel	le nombre de lettre en moins toléré pour trouver une anagramme

Le programme en lui-même

Deux parties.

- Première partie

Vous lancez votre programme avec un seul paramètre. Ce dernier vous retourne toutes les anagrammes parfaites du mot rentré en paramètre.

- Deuxième partie

Vous lancez votre programme avec deux paramètres. Ce dernier vous retourne toutes les anagrammes du mot rentré en paramètre avec le nombre de lettre *nb* en moins. Inspirer vous des exemples ci-dessous pour mieux comprendre ce qui vous est demandé de faire.

Le mot rentré en paramètre **n'est pas obligatoirement** un mot du dictionnaire fourni.



Attention, à bien gérer toutes les erreurs possibles.

Quelques exemples

Voici quelques exemples, préalablement vérifiés et correctement écrit, qui permettront de guider votre réflexion quant aux résultats à fournir.

Pour l'**exercice 1**, avec l'affichage de votre choix, vous devez retrouver ce résultat.

```
[Montblanc:~ MontBlanc$ php anagram.php marie
Array
(
    [0] => aimer
    [1] => maire
    [2] => ramie
)
exec time: 0.91s
```

```
[Montblanc:~ MontBlanc$ php anagram.php tarte
Array
(
    [0] => taret
)
exec time: 0.9s
```

Pour l'**exercice 2**, avec l'affichage de votre choix, vous devez retrouver ce résultat.

```
[Montblanc:~ MontBlanc$ php anagram.php marie 0
Veuillez rentrer un nombre correct
exec time: 0s
```

```
[Montblanc:~ MontBlanc$ php anagram.php marie 1
Array
(
    [0] => aime
    [1] => aire
    [2] => amer
    [3] => amie
    [4] => arme
    [5] => maie
    [6] => mare
    [7] => mari
    [8] => mira
    [9] => mire
    [10] => raie
    [11] => rame
    [12] => rami
    [13] => rima
    [14] => rime
)
exec time: 1.19s
```

```
Montblanc:~ MontBlanc$ php anagram.php marie 2
Array
(
    [0] => aie
    [1] => air
    [2] => ami
    [3] => are
    [4] => ira
    [5] => ire
    [6] => mai
    [7] => mer
    [8] => mie
    [9] => mir
    [10] => rai
    [11] => rem
    [12] => ria
    [13] => rie
)
exec time: 1.61s
```

```
Montblanc:~ MontBlanc$ php anagram.php marie 6
Veuillez rentrer un nombre moins grand que la taille du mot
exec time: 0s
```

```
Montblanc:~ MontBlanc$ php anagram.php marie -1
Veuillez rentrer un nombre correct
exec time: 0s
```

```
Montblanc:~ MontBlanc$ php anagram.php tarte 1
Array
(
    [0] => rate
    [1] => tare
)
exec time: 1.16s
```

```
Montblanc:~ MontBlanc$ php anagram.php tarte 2
Array
(
    [0] => are
    [1] => art
    [2] => rat
)
exec time: 1.59s
```


Restriction

Les seules fonctions autorisées sont :

- count()
- strlen()
- isset()
- fopen()
- fgets()
- fclose()
- echo()
- print_r()
- microtime()
- round()

Toutes les autres fonctions sont **INTERDITES !!!**



Un nom respect d'une de ces fonctions entraînera un -42.

Bonus

Libre à vous de faire les bonus qui vous semble adapté à votre programme. Pour vous donner quelques pistes :

- Directement, avec une option supplémentaire, vous avez tous les anagrammes possibles (parfait ou non), classé par ordre d'anagrammes parfait, c'est à dire qu'au début vous listez les anagrammes parfaites, puis avec une lettre en moins, puis deux, puis trois ...
- Vous affichez le temps de calcul à la fin de chaque opération
- Vous pouvez rajouter un mot au dictionnaire à l'aide d'une option, sans à avoir à ouvrir le dictionnaire et sans rien effacer au contenu du dictionnaire présent
- Vous récupérez un dictionnaire en anglais et faites une option qui permet d'utiliser ce même dictionnaire pour découvrir des anagrammes dans cette langue.
- Un meilleur affichage des résultats
- Essayez de battre nos temps de calcul :)

Bon courage à tous.