

RÉPUBLIQUE TUNISIENNE ☆☆☆ Ministère de l'Éducation	EXAMEN DU BACCALAURÉAT	SESSION 2023
	ÉPREUVE PRATIQUE D'INFORMATIQUE	
	Sections : Mathématiques, Sciences expérimentales et Sciences techniques	
	Coefficient de l'épreuve : 0.5	Durée : 1h

Important :

1. Une solution modulaire au problème posé est exigée.
2. Dans le répertoire **Bac2023**, créez un dossier de travail ayant comme nom votre numéro d'inscription (6 chiffres) et dans lequel vous devez enregistrer, au fur et à mesure, tous les fichiers solution à ce sujet.

Triage de mots

On se propose de concevoir une interface graphique permettant de saisir une chaîne de caractères **ch** contenant aux maximum 15 mots puis de trier les mots qui la constituent dans l'ordre croissant de leurs longueurs.

Exemple :

Pour **ch** = "une hirondelle ne fait pas le printemps."

Après le triage des mots de **ch**, on obtient le résultat suivant : "ne le une pas fait printemps hirondelle"

N.B. Le point "." indiquant la fin de la chaîne, ne fait pas partie du dernier mot. Par conséquent ne figure pas dans la chaîne cryptée.

L'interface graphique à concevoir contient les éléments suivants, comme l'illustre la capture d'écran ci-dessous :

- Un label contenant le texte "**Triage de mots**"
- Un label contenant le texte "**Introduire une phrase :** "
- Une zone de saisie pour la saisie d'une phrase
- Un label pour afficher le résultat
- Un bouton intitulé "**Trier**"

Travail demandé :

- 1) Concevoir l'interface graphique présentée précédemment et l'enregistrer sous le nom **InterfaceTriage**.
- 2) Créer un programme Python et l'enregistrer sous le nom **Triage**, dans lequel, il est demandé :
 - a) de développer une fonction nommée **Trier (ch)** qui permet de retourner une chaîne de caractères formée par les mots de la chaîne de caractères **ch** triés dans l'ordre croissant de leurs longueurs.
 - b) de développer un module **Play**, qui s'exécute suite à un clic sur le bouton "**Trier**", permettant :
 - de récupérer la chaîne **ch** saisie. La chaîne **ch** doit être non vide et de taille inférieure à 50. Elle commence obligatoirement par une lettre, se termine par un point et chaque deux mots consécutifs sont séparés par un seul espace.
 - de trier les mots de la chaîne **ch** en utilisant la fonction **Trier (ch)** puis d'afficher le résultat via le **label** dédié à l'affichage dans l'interface graphique **InterfaceTriage**.
 - c) d'ajouter les instructions permettant d'exploiter l'interface graphique intitulée **InterfaceTriage** en se référant à l'annexe ci-après.

N.B. : l’affichage doit être conforme aux exemples d’exécutions suivants :

Exemples d’exécutions :

Triage de mots

Introduire une phrase :

Veuillez introduire une phrase

Trier

Triage de mots

Introduire une phrase :

Entre 2 mots un seul espace est autorisé

Trier

Triage de mots

Introduire une phrase :

La chaîne doit se terminer par un point

Trier

Triage de mots

Introduire une phrase :

ne le une pas fait printemps hirondelle

Trier

Annexe

```
from PyQt5.uic import loadUi
from PyQt5.QtWidgets import QApplication
.....
.....
app = QApplication([])
windows = loadUi ("Nom_Interface.ui")
windows.show()
windows.Nom_Bouton.clicked.connect (Nom_Module)
app.exec_()
```

Kiteb.net: le site web éducatif

Grille d’évaluation

Tâches	Nombre de points
Conception de l’interface " InterfaceTriage "	4 pts
Création et enregistrement du programme " Triage "	1 pt
Développement de la fonction " Trier "	6 pts
Développement du module " Play "	4 pts
Ajout des instructions de l’exploitation de l’interface	3 pts
Modularité et cohérence	2 pts