

NOUVELLE CALÉDONIE – 2023 – sujet2

Exercice 2 (4 points)

Tous les programmes de cet exercice seront écrits en langage Python.

- 1) Un mot est un palindrome lorsqu'il s'écrit de la même façon à l'endroit et à l'envers. Exemples : radar, abcdcba, kayak, laval, non, ressasser, elle.
Soit la fonction ci-dessous appelée palindrome.

```
def palindrome(mot):  
    n=len(mot)  
    est_palindrome=True  
    for k in range(n//2):  
        if mot[k]!=mot[n-k-1]:  
            est_palindrome=False  
    return est_palindrome
```

- a. **Écrire** les spécifications, sous la forme d'une chaîne de documentation (docstring), de cette fonction. Cette documentation précisera l'entrée et son type, la sortie et son type, le rôle de cette fonction.
- b. **Indiquer** sans justification la complexité en temps de calcul de cet algorithme pour un mot de longueur n.
Remarque : on ne comptera que les comparaisons de caractères.
- c. **Indiquer** ce qu'est une fonction récursive.
- d. **Écrire et compléter**, sur la copie, les lignes 2, 4 et 5 de la fonction récursive appelée `palindrome_recursive`, qui prend en paramètre un mot et qui teste si ce mot est un palindrome.

```
1. def palindrome_recursive(mot):  
2.     if len(mot)<=.....:  
3.         return True  
4.     if mot[0]!=mot[.....]:  
5.         return .....  
6.     return palindrome_recursive(mot[1:len(mot)-1])
```

Remarque : l'expression `chaîne[debut:fin]` donne la sous-chaîne comportant les caractères de l'indice `debut` inclus à l'indice `fin` exclu.

- 2) On s'intéresse à une version simplifiée du jeu télévisé Motus. Le jeu repose sur la recherche de mots d'un nombre fixé de lettres. Un candidat propose un mot et doit l'épeler. Le mot doit être correctement orthographié, sinon il est refusé. On s'intéresse uniquement aux lettres bien placées. Un mot mystère est défini au préalable. Sa première lettre est donnée et ses lettres restantes

signalées par des tirets. Le programme invite alors le candidat à saisir un mot.

Par exemple :

- Le mot mystère est MOMIE.
- Le programme va d'abord afficher « M----- ».
- Le candidat est ainsi invité à chercher un mot de cinq lettres commençant par M.
- Le candidat propose le mot MINCE.

Les lettres M et E sont bien placées dans le mot à deviner MOMIE, la console affiche M-- E.

- a. **Écrire** la fonction `test_mot` qui prend en paramètres le mot à deviner en majuscules et un mot proposé en majuscules par le joueur. Cette fonction renvoie le mot formé des lettres correctement placées et de tirets ailleurs comme dans l'exemple ci-dessus. On suppose que le mot proposé par le joueur a le bon nombre de lettres.

Par exemple :

```
>>> test_mot('MOMIE', 'MINCE')
'M---E'
```

- b. On a rédigé un programme `jeu_motus` qui prend en paramètre le mot à deviner et qui :

- affiche la première lettre du mot à deviner suivie de tirets ;
- affiche la phrase « donner votre proposition : » qui invite le candidat à entrer une proposition ;
- affiche les lettres bien placées et des tirets ailleurs ;
- recommence jusqu'à ce que le candidat ait gagné ou qu'il ait atteint le nombre maximal d'essais autorisés, ce nombre maximal étant égal au nombre de lettres du mot ;
- affiche le message « Félicitations » si le candidat a gagné ;
- donne le mot mystère si le candidat a perdu.

Par exemple, on souhaite obtenir les affichages suivants :

```
>>>jeu_motus('MOTUS')
M----
donner votre proposition : MATIN
M-T--
donner votre proposition : MATHS
M-T-S
donner votre proposition : MOTOS
MOT-S
donner votre proposition : MOTUS
Félicitations, vous avez gagné en 4 coups.
```

Expliquer pourquoi le programme suivant ne donne pas toujours le résultat attendu.

```
def jeu_motus(mot):
    n=len(mot)
    nb_coups=0
    gagne=False
    print(mot[0]+'-'*(n-1))
    while nb_coups<n and gagne==False:
        mot_joueur=input('donner votre proposition : ')
        if mot_joueur==mot:
            gagne=True
        else:
            print(test_mot(mot, mot_joueur))
    if gagne:
        print('Félicitations, vous avez gagné en ', nb_coups, 'coups.')
    else:
        print('perdu, il fallait trouver : ',mot)
```