

CONTROLE PARTIEL D'ALGORITHMIQUE

1H30 – TOUS DOCUMENTS AUTORISES, SAUF SUPPORTS NUMERIQUES

*Vous aurez à produire 2 (et seulement 2) fichiers : le fichier principal **Palindrome.py** contenant l'application et le fichier module **Fonctions_Palin.py** qui contiendra l'ensemble de fonctions que vous écrirez.*

Le nom des fonctions ainsi que le nom de certaines variables sont imposés. Vous devez les respecter.

SUJET : PALINDROME

On se propose d'écrire un programme qui permet de déterminer si un nombre saisi par l'utilisateur est un palindrome, à savoir que sa lecture de gauche à droite est identique à celle de sa lecture de droite à gauche

Exemple de palindromes :

- 393 est un palindrome d'un nombre à 3 chiffres
- 8448 est un palindrome d'un nombre à 4 chiffres
- 951159 est un palindrome d'un nombre à 6 chiffres
- 9849 n'est pas un palindrome (la lecture de droite à gauche donne 9489)

ETAPE 1 : DETERMINATION DE LA TAILLE DU NOMBRE

Afin de ne pas compliquer le programme l'application va se limiter au nombre contenant entre 3 et 6 chiffres.

◇ Ecrire dans le module **Fonctions_Palin.py** la fonction **Taille** qui retournera le nombre de chiffre qui contient un nombre entier passé en argument d'entrée. Si ce nombre est inférieur à 3 ou supérieur à 6 la fonction retournera la valeur -1

Exemple d'appel :

- *Taille(1234) retournera la valeur 4*
- *Taille(60245) retournera la valeur 5*
- *Taille(75) retournera la valeur -1 (valeur trop petite)*
- *Taille(9857457) retournera la valeur -1 (valeur trop grande)*

◇ Ecrire le début de l'application dans le fichier principal **Palindrome.py** qui demandera à l'utilisateur de donner un chiffre. L'application appellera ensuite la fonction **Taille** et affichera la valeur retournée par la fonction.

- La variable qui contiendra le chiffre donné par l'utilisateur se nommera obligatoirement **IsPalin**.
- La variable qui contiendra la taille de **IsPalin** se nommera obligatoirement **Howmuch**.

ETAPE 2 : DETERMINATION DE LA TAILLE

◇ Recopier dans le module **Fonctions_Palin.py** la fonction **Extract** suivante :

```
def Extract(Nb,digit) :  
    Val1 = (Nb+0.01)/(10**(digit))  
    Val2 = Val1 - math.trunc(Val1)  
    Val3 = math.trunc(Val2*10)  
    return (Val3)
```

◇ Complétez **Palindrome** pour vérifier que cette fonction permet d'extraire par exemple le deuxième digit du nombre passé en premier argument puis le quatrième digit.

Sur la notion de digit : dans le nombre 78594, le premier digit vaut 4, le deuxième digit vaut 9, le troisième digit vaut 5, le quatrième digit vaut 8 et le cinquième digit vaut 7

ETAPE 3 : EXTRACTION DES CHIFFRES EN « MIROIR »

Il s'agit maintenant de récupérer 2 chiffres positionnés en miroir afin de pouvoir (par la suite) les comparer et déterminer si le nombre est un palindrome.

Exemples :

- Avec la valeur 75987 il faudra récupérer le premier digit (7) et le cinquième digit (7) puis le deuxième (8) et le quatrième digit (5).
 - Avec la valeur 4554 il faudra récupérer le premier digit (4) et le quatrième digit(4) puis le deuxième (5) et le troisième digit (5).
- ◇ Ecrire la fonction **Deux_a_Deux(Val,digit,Dim)** qui prendra en argument d'entrée le nombre à tester(**Val**), un digit à extraire (**Digit**), et la taille du chiffre (**Dim**), et qui retournera les deux valeurs attendues à savoir le chiffre de position « digit » et celui placé en miroir par rapport au « milieu » du nombre. La taille est passée en argument d'entrée afin de pouvoir « calculer » la position du second digit à extraire.

Exemples :

- **Deux_a_Deux(9587,2,4)** retournera les valeur 8 (le deuxième digit) et 5 (le troisième digit, celui symétrique au deuxième digit par rapport au milieu)
 - **Deux_a_Deux(128475,1,6)** retournera les valeur 5 (le premier digit) et 1 (le sixième digit), celui symétrique au premier digit par rapport au milieu)
- ◇ Ecrire dans le fichier **Palindrome** les lignes qui permettent d'appeler cette fonction pour les deux exemples présentés ci-dessus et d'afficher les valeurs alors retournées.

ETAPE 4 : FINALISATION

- ◇ Finaliser l'application pour déterminer si le nombre saisi est ou n'est pas un palindrome. La vérification ne se fera que si le nombre saisi est « acceptable » c'est-à-dire constitué de 3 à 6 chiffres.

Laissez dans le code les lignes de test correspondant aux étapes 1, 2 et 3