

Situations

Caractères & Codes

On donne le programme suivant :

Python

```
num = 69
car = "E"
print("Caractère :", chr(num))
print("Code :", ord(car))
```

1. Copier/coller, puis tester le programme. Qu'est ce qu'il affiche ?

2. Modifier les valeurs de `num` et de `car` :

- o `num = 101` et `car = "e"`
- o `num = 52` et `car = "4"`
- o `num = 90` et `car = "Z"`
- o `num = 33` et `car = "!"`

3. En déduire le rôle des fonctions `chr(num)` et `ord(car)`.

4. En se référant à la table ASCII suivante :

Trouver quels sont les caractères dont le code est :

- o 35
- o 62
- o 59

Trouver quels sont les codes des caractères :

- o "+"
- o ")"
- o "z"

Nombre automorphe

Un **nombre automorphe** est un entier naturel dont la suite des chiffres du carré se termine par celle du nombre lui-même. Par exemple,

$$5^2 = 25, 6^2 = 36, 25^2 = 625,$$
$$76^2 = 5776, 890625^2 = 793212890625$$

On demande d'écrire un programme qui teste si un nombre est automorphe. On donne, pour cela, le programme incomplet suivant :

Python

```
# Todo 1 : Enlever le commentaire # de la ligne suivante
# n = int(input("Donner un nombre > 0 ? "))
# Todo 2 : à supprimer
n = 90625
# Todo 3 : calculer le carré de n
nc = 8212890625
# Todo 4 : convertir n et nc en chaînes de caractères
nch = "90625"
ncch = "8212890625"
# Todo 5 : trouver la longueur de nch et de ncch
l1 = 5
l2 = 10
# Todo 6 : utiliser l1 et l2 pour retrouver les l1 derniers caractères de ncch
dcar = "90625"
# Todo 7 : déterminer si n est automorphe
automorphe = True
print(n, "est automorphe ?", automorphe)
```

On demande de :

1. Déterminer, d'après **la définition d'un nombre automorphe**, quels sont les nombres automorphes dans la liste suivante :
 - o 376
 - o 9376
 - o 17
 - o 125
2. Calculer le carré de **n**, **Todo 3**, puis afficher la valeur de **nc**.

Python

```
print(n, "^ 2 =", nc)
```

3. Convertir **n** et **nc** en chaînes, **Todo 4**, et afficher leurs types.

Python

```
print(nch, "est de type", type(nch))
print(ncch, "est de type", type(ncch))
```

4. Trouver les longueurs de **nch** et de **ncch**, **Todo 5**, puis afficher le nombre de chiffres de **n** et de **nc**.

Python

```
print(n, "contient", l1, "chiffres")
print(nc, "contient", l2, "chiffres")
```

5. Afficher le dernier chiffre de **ncch**.

Python

```
print("Le dernier chiffre de", ncch, "est", "???) # remplacer "???" par le code adéquat
```

6. Afficher les deux derniers chiffres de **ncch**.

Python

```
print("Les deux derniers chiffres de", ncch, "sont", "???) # remplacer "???" par le code adéquat
```

7. Retrouver les **l1** derniers chiffres de **ncch**, **Todo 6**.

Python

```
print("Les", l1, "derniers chiffres de", ncch, "sont", "???) # remplacer "???" par le code adéquat
```

8. Afficher le premier chiffre de **nch** ?

Python

```
print("Le premier chiffre de", ncch, "est", "???) # remplacer "???" par le code adéquat
```

9. Afficher les deux premiers chiffres de **nch** ?

Python

```
print("Les deux premiers chiffres de", ncch, "est", "???) # remplacer "???" par le code adéquat
```

10. Tester si le nombre **7** est automorphe ? Qu'affiche le programme ?
11. Tester si le nombre **36** est automorphe ? Qu'affiche le programme ? Quel est le problème ?
12. Répondre à **Todo 7**.
13. Compléter le programme **Todo 2**, puis **Todo 1**.

Le plus âgé