STRUCTURES DE DONNEES – DEVOIR

EXERCICE 1: LA POO EN QUELQUES QUESTIONS

1) On définit la classe Identite de la manière suivante:

```
1 class Identite:
2     def __init__(self, nom, prenom, an):
3         self.nom=nom
4         self.prenom=prenom
5         self.an=an
6     def age(self, a):
7         return a-self.an
```

Qu´affichera la console à la suite de l´exécution des instructions suivantes? Justifiez toutes vos réponses.

a)

```
maria=Identite('Maria','Pires',1999)
print(maria.nom +' '+  maria.prenom)
```

b)

```
maria=Identite('Maria','Pires',1999)
print(maria.nom +' '+  maria.an)
```

c)

```
hugo=Identite('Hugo','Dupont',2004)
print(hugo.age(2021))
```

2) Ecrire le code de la classe Voiture qui permet d´afficher *'Ferrari Rouge'* après avoir saisi les instructions suivantes:

```
>>> F430=Voiture('Ferrari','rouge')
>>> print(F430)
Ferrari rouge
```

3) Expliquez en détail ce que permet d´afficher ce programme:

```
import random

class Piece:
    def alea(self):
        return random.randint(0,1)
    def moyenne(self,n):
        tirage=[]
        for i in range (n):
            tirage.append(self.alea())
        return sum(tirage)/n
p=Piece()
print(p.moyenne(100))
```

Rappel: la fonction sum calcule la somme des termes d´une liste.

EXERCICE 2: LA CLASSE Date

On construit le code d'une variable globale et d'une classe Date représenté ci-dessous:

```
liste = ["janvier", "février", "mars", "avril", "mai", "juin", "juillet", "aout",
"septembre", "octobre", "novembre", "décembre"]
class Date:
    def __init__(self, jour, mois, annee):
            self.jour = jour
            self.mois = mois
            self.annee = annee
    def __lt__(self, d):
        if self.annee < d.annee:
            return True
        elif self.annee > d.annee:
            return False
        else:
            if self.mois < d.mois:
                return True
            elif self.mois > d.mois:
                return False
            else:
                return self.jour < d.jour
```

La méthode __lt__ est la méthode spéciale de comparaison ('lower than').

1) Qu´affichera la console à l´exécution des instructions suivantes? Justifiez votre réponse.

```
d1 = Date(17, 12, 2021)
d2 = Date(11, 12, 2021)
print(d1<d2)
```

2) Ecrire le code d'une méthode de la classe Date qui permet d´afficher *'17décembre2021'* après avoir saisi les instructions suivantes:

```
>>> d1 = Date(17, 12, 2021)
>>> print(d1)
17décembre2021
```

EXERCICE 3 : UNE HISTOIRE DE PILE

Dans cet exercice, on considère une pile d'entiers positifs. On suppose que les quatre fonctions suivantes ont été programmées préalablement en langage Python :

`empiler(P, e)` : ajoute l'élément e sur la pile P ;
`depiler(P)` : enlève le sommet de la pile P et retourne la valeur de ce sommet ;
`est_vide(P)` : retourne True si la pile est vide et False sinon ;
`creer_pile()` : retourne une pile vide.

Dans cet exercice, seule l'utilisation de ces quatre fonctions sur la structure de données pile est autorisée.

1) **Compléter le schéma ci-dessous.** Plus précisément, pour chaque colonne :

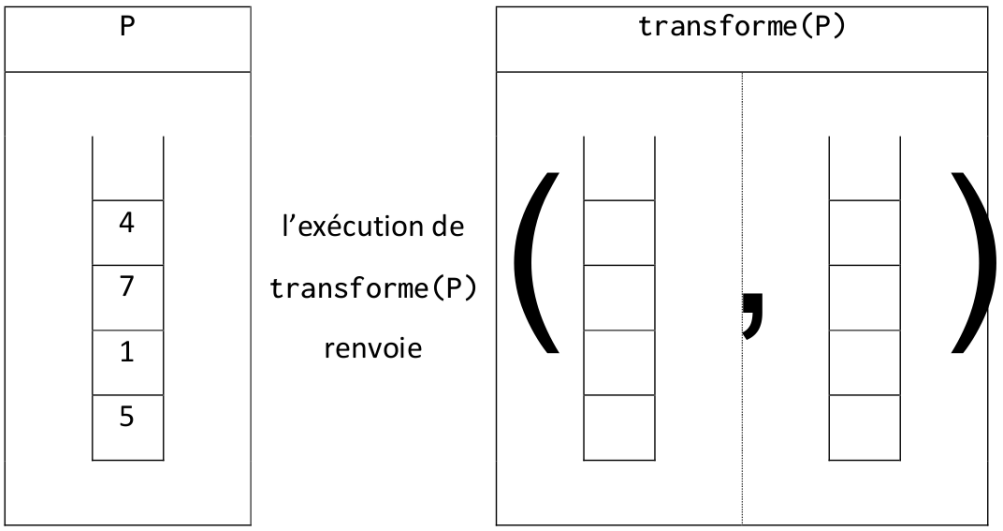
- écrire l'état de la pile dans la 2e ligne après l'exécution de l'instruction de la 1ère ligne
- écrire ce que renvoie la fonction utilisée dans la 3e ligne après l'exécution de l'instruction de la 1ère ligne (indiquer « None » si la fonction ne retourne aucune valeur)

	Etape 0 Pile d'origine P	Etape 1 empiler(P,8)	Etape 2 depiler(P)	Etape 3 est_vide(P)
				
Retour de la fonction		.....	.....	.....

2) On propose la fonction ci-dessous, qui prend en argument une pile P et renvoie un couple de piles :

```
def transforme(P) :
    Q = creer_pile()
    while not est_vide(P) :
        v = depile(P)
        empile(Q,v)
    return (P,Q)
```

a. **Compléter sur le sujet le document ci-dessous** :

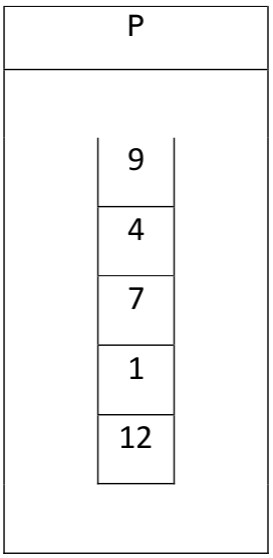


3) **Écrire une fonction en langage Python maximum(P) recevant une pile P comme argument et qui renvoie la valeur maximale de cette pile. Après exécution de la fonction, la pile P devra être dans son état initial.**

4) On souhaite connaître le nombre d'éléments d'une pile à l'aide de la fonction taille(P), sans modifier la pile P.

a. Proposer une stratégie écrite en langage naturel et/ou expliquée à l'aide de schémas, qui permette de mettre en place une telle fonction.

b. Donner le code Python de cette fonction taille(P) (on pourra utiliser les cinq fonctions déjà programmées).



taille(P) retournera donc l'entier 5