

Erreur avec arrêt du programme

Ces erreurs génèrent l'affichage du *traceback* dans la console. Ce qui permet d'aider à corriger celle-ci.

1.1 Affectation simple

```
1 # a == 10
2 a = 10
```

1.2 Affectation multiple

Cette affectation multiple ne fonctionne pas, corrigez la.

```
1 # a, b = 10
2 a, b = 10, 10
```

1.3 Opérateur d'affectation mal employé

Corriger le script suivant

```
1 """
2 a = 0
3 if a = 0:
4     print('a est nul')
5 """
6 a = 0
7 if a == 0:
8     print('a est nul')
```

1.4 Structure avec bloc

Corrigez les scripts suivants

```
1 """
2 a = 17
3 if a < 18
4     print('vous n'avez pas le droit de vote')
5 """
6 a = 17
7 if a < 18:
8     print('vous n'avez pas le droit de vote')
```

```
1 """
2 for i in range(10)
3     print('i')
4 """
5 for i in range(10):
6     print(i)
```

```
1 0
2 1
3 2
```

```

4 3
5 4
6 5
7 6
8 7
9 8
10 9

```

```

1 """
2 note = 20
3 if note > 10:
4     print('vous avez le bac')
5 else:
6     print('vous avez echoué')
7 """
8 note = 20
9 if note > 10:
10     print('vous avez le bac')
11 else:
12     print('vous avez echoué')

```

vous avez le bac

1.5 Fonctions

corrigez les scripts suivants

```

1 """
2 def aire(L,h):
3     A = L * h
4     return a
5 print(def aire 3,4)
6 print(aire 3 4)
7 aire(12)
8 """
9 def aire(L,h):
10     a = L * h
11     return a
12 print(aire(3,4))
13 aire(3,4)

```

12

```

1 """
2 volume(a,b,c):
3     return A * B * C
4 """
5 def volume(a,b,c):
6     return a * b * c
7 volume(12,5,2)

```

120

1.6 Chaines de caractères et entiers

rappel : chaine de caracteres => entier : fonction `int`

entier => chaine de caracteres : fonction `str`

```

1  """
2  a = 'Nous partimes à '
3  b = 500
4  c = 'Mais par un prompt renfort, nous nous vimes trois mille en arrivant au
   port'
5  print(a + b + c)
6  """
7  a = 'Nous partiment à '
8  b = str(500)
9  c = ' Mais par un prompt renfort, nous vimes trois mille en arrivant'
10 print(a + b + c)

```

Nous partiment à 500 Mais par un prompt renfort, nous vimes trois mille en arrivant

```

1  # Ce script devrait afficher 2021 et il affiche ...23. Corrigez le
2  """
3  a = 2
4  b = '0'
5  c = '21'
6  print(a+int(b)+int(c))
7  """
8  a = 2
9  b = '0'
10 c = '21'
11 print(str(a) + b + c)

```

2021

```

1  """
2  a = 18
3  b = '2'
4  print(a/b)
5  """
6  a = 18
7  b = '2'
8  print(a/int(b))

```

9.0

On souhaite afficher tous les nombres de 0 à 7, comment faut-il corriger le script pour y parvenir?

```

1  """
2  for i in range(7):
3      print('i')
4  # affiche i sept fois
5  """
6  for i in range(7):
7      print(i)

```

```

1  0
2  1
3  2
4  3
5  4
6  5
7  6

```

On souhaite afficher à chacun des appels de la fonction `bonjour` le message : *Bonjour Paul*. Corriger les instructions proposées dans ce script.

```

1  prenom = 'Paul'

```

```
2 def bonjour(name):
3     return 'bonjour ' + name
4
5     """
6     print(bonjour(Paul))
7     print(bonjour('prenom'))
8     print(bonjour('name'))
9     print(bonjour(name))
10    print(bonjour())
11    """
12    print(bonjour('Paul'))
13    print(bonjour(prenom))
```

Résultat :

```
1 bonjour Paul
2 bonjour Paul
```

1.7 Import de module

On rappelle que le module `math` s'importe avec l'instruction : `import math`

`pi` est une constante du module `math` et s'appelle avec `math.pi` UNE FOIS le module importé.

`sin` est une fonction du module `math` et s'appelle avec `math.sin`

```
1     """
2     R = 1
3     L = 2 * pi * R
4     """
5     import math
6     R = 1
7     L = 2 * math.pi * R
8     print(L)
```

6.283185307179586

```
1     """
2     angle = 30
3     radian = angle * math.pi/180
4     sin(radian)
5     """
6     angle = 30
7     radian = angle * math.pi/180
8     math.sin(radian)
```

0.49999999999999994

Partie 2

Erreur de structure

Ces erreurs entraînent un fonctionnement du programme différent de ce qui est attendu. Leur recherche et correction est souvent plus difficile.

2.1 Problème 1 : Structure conditionnelle

On souhaiterait que le programme affiche :

```

1 vous avez 0 ans , vous êtes à la crèche
2 vous avez 1 ans , vous êtes à la crèche
3 vous avez 2 ans , vous êtes à la crèche
4 vous avez 3 ans , vous êtes en primaire
5 vous avez 4 ans , vous êtes en primaire
6 vous avez 5 ans , vous êtes en primaire
7 vous avez 6 ans , vous êtes en élémentaire
8 vous avez 7 ans , vous êtes en élémentaire
9 vous avez 8 ans , vous êtes en élémentaire
10 vous avez 9 ans , vous êtes en élémentaire
11 vous avez 10 ans , vous êtes en élémentaire
12 vous avez 11 ans , vous êtes en élémentaire
13 vous avez 12 ans , vous êtes au collège

```

Le script suivant devrait avantageusement utiliser les structures conditionnelles avec alternatives pour parvenir à cet affichage. Corrigez le.

```

1 """
2 for age in range(13):
3     if age < 3:
4         print('vous avez ' + str(age)+ ' ans, vous êtes à la crèche')
5     if age < 6:
6         print('vous avez ' + str(age)+ ' ans, vous êtes en primaire')
7     if age < 12:
8         print('vous avez ' + str(age)+ ' ans, vous êtes en élémentaire')
9     if age < 16:
10        print('vous avez ' + str(age)+ ' ans, vous êtes au collège')
11 """
12 for age in range(13):
13     if age < 3:
14         print('vous avez ' + str(age)+ ' ans, vous êtes à la crèche')
15     elif age < 6:
16         print('vous avez ' + str(age)+ ' ans, vous êtes en primaire')
17     elif age < 12:
18         print('vous avez ' + str(age)+ ' ans, vous êtes en élémentaire')
19     elif age < 16:
20         print('vous avez ' + str(age)+ ' ans, vous êtes au collège')

```

Résultat :

```

1     vous avez 0 ans , vous êtes à la crèche
2     vous avez 1 ans , vous êtes à la crèche
3     vous avez 2 ans , vous êtes à la crèche
4     vous avez 3 ans , vous êtes en primaire
5     vous avez 4 ans , vous êtes en primaire
6     vous avez 5 ans , vous êtes en primaire
7     vous avez 6 ans , vous êtes en élémentaire
8     vous avez 7 ans , vous êtes en élémentaire
9     vous avez 8 ans , vous êtes en élémentaire
10    vous avez 9 ans , vous êtes en élémentaire
11    vous avez 10 ans , vous êtes en élémentaire
12    vous avez 11 ans , vous êtes en élémentaire
13    vous avez 12 ans , vous êtes au collège

```

2.2 Problème 2 : Utiliser l'opérateur % ou l'opérateur //

On cherche à numéroter les jours du mois. On suppose que l'année ne contient que des mois de 30 jours. L'affichage doit être :

```

1  debut
2  jour n°1
3  jour n°2
4  ...
5  jour n°30
6  jour n°1
7  ...
8  jour n°10
9  fin

```

Modifier le programme pour qu'il affiche la bonne sequence. Les mots *debut* et *fin* ne devront être affichés qu'en debut et fin de sequence.

```

1  """
2  for i in range(41):
3      print('debut')
4      print(i//30)
5      print('fin')
6  """
7  print('debut')
8  for i in range(41):
9      print(i % 30)
10 print('fin')

```

```

1  debut
2  0
3  1
4  2
5  3
6  4
7  5
8  6
9  7
10 8
11 9
12 10
13 11
14 12
15 13
16 14
17 15
18 16
19 17
20 18
21 19
22 20
23 21
24 22
25 23
26 24
27 25
28 26
29 27
30 28
31 29

```

```

32 0
33 1
34 2
35 3
36 4
37 5
38 6
39 7
40 8
41 9
42 10
43 fin

```

2.3 Problème 3 :

corriger la fonction suivante pour qu'elle retourne bien la valeur 30° lorsqu'on l'appelle avec arcsin(0.5)

Rappel :

$$angle_{deg} = \frac{angle_{radian} * 180}{PI}$$

```

1  """
2  import math
3  sinus_de_l_angle = 0.5
4  def arcsin(x):
5      a = math.asin(x)
6      return a
7  """
8  import math
9  sinus_de_l_angle = 0.5
10 def arcsin(x):
11     """calcule angle à partir de son sinus"""
12     a = math.asin(x) * 180 / math.pi
13     return a
14
15 arcsin(sinus_de_l_angle)

```

```

1 29.999999999999996

```