

Devoir Première NSI n°1	
Base Python et numération	
Première – spécialité NSI	Octobre 2020
Durée 2h	
Memo Python en annexe autorisé, Calculatrice interdite	

NOM :	PRENOM :	CLASSE :
-------	----------	----------

Pour répondre au QCM, indiquer votre réponse dans le cadre à **droite**

Exercice 1 : QCM

1. Quel est l'opérateur qui permet de tester une égalité entre deux variables ?

- A - =
- B - <>
- C - ==
- D - !=

C

On donne le script Python suivant :

```
def reste_division(nb:int, diviseur: int):
    """Renvoie le reste de la division euclidienne
    de 'nb' par 'diviseur'"""
    reste = nb % diviseur
    if reste < 0:
        reste = reste + abs(diviseur)
    return reste
```

2. Quel est le rôle de `reste_division` ?

- A - C'est le nom de la fonction
- B - C'est un paramètre
- C - C'est un mot clé
- D - C'est une variable locale

A

3. Quel est le rôle de `reste` ?

- A - C'est le nom de la fonction
- B - C'est un paramètre
- C - C'est un mot clé
- D - C'est une variable locale

D

4. Quel est le rôle de `diviseur` ?

- A - C'est le nom de la fonction
- B - C'est un paramètre
- C - C'est un mot clé
- D - C'est une variable locale

B

5. En écriture binaire on utilise les symboles :

- A - 1 et 2
- B - 0 et 1
- C - -1 et 1
- D - 0 et 10

B

6. Pour représenter des entiers (sans ordinateur), les bases possibles sont

- A - Seulement 2 et 10
- B - seulement 2, 10 et 16
- C - autant que l'on peut imaginer de signes

C

Devoir Première NSI n°1	
Base Python et numération	
Première – spécialité NSI	Octobre 2020
Durée 2h	
Memo Python en annexe autorisé, Calculatrice interdite	

7. Un groupement de 8 bits s'appelle :

- A - une brochette
- B - un quintet
- C - un octet
- D - un Mégabit

C

8. On peut coder sur un octet :

- A - 256 valeurs
- B - 255 valeurs
- C - 8 valeurs
- D - 1000 valeurs

A

9. Combien de mots binaires différents peut-on former avec 3 bits?

- A - 7
- B - 8
- C - 2^3-1
- D - 2^3+1

B

10. Dans un octet le bit le plus à gauche peut être nommé :

- A - bit de poids fort
- B - bit de poids faible
- C - bit des unités
- D - bit principal

A

11. Si on rencontre le nombre 3F5, dans quel système de numération est-on ?

- A - Binaire
- B - Décimal
- C - Octal
- D - Hexadécimal

D

12. Quelle est la représentation binaire, en complément à 2 sur 8 bits, de l'entier négatif -25 ?

- A - 0001 1001
- B - 1001 1001
- C - 1110 0110
- D - 1110 0111

D

13. Quelle est la représentation hexadécimale de l'entier dont la représentation binaire s'écrit 0100 1001 1101 0011 ?

- A - 18899
- B - 3D94
- C - 49D3
- D - 93A3

C

14. Le nombre 3F correspond au décimal :

- A - 63
- B - 48
- C - 45
- D - 18

A

15. L'écriture binaire de $(6,625)_{10}$ est :

- A - 110,101
- B - 0,101
- C - 110,101101...

A

Devoir Première NSI n°1	
Base Python et numération	
Première – spécialité NSI	Octobre 2020
Durée 2h	
Memo Python en annexe autorisé, Calculatrice interdite	

16. Parmi les nombres suivants, écrits en base 10, quel est celui qui a une écriture binaire infinie en base 2 ?

A - 1,25

B - 0,75

C - 1,7

C

Exercice 2 :

1 - Réaliser l'addition en base 2

$$\begin{array}{r}
 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0 \\
 + 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1 \\
 \hline
 00100011
 \end{array}$$

2 - Entiers signés en base 2

00010101 est

X positif

ou

☐ négatif

 et a pour opposé : 11101011

01010011 est

X positif

ou

☐ négatif

 et a pour opposé : 10101101

11011010 est

☐ positif

ou

X négatif

 et a pour opposé : 00100110

Devoir Première NSI n°1	
Base Python et numération	
Première – spécialité NSI	Octobre 2020
Durée 2h	
Memo Python en annexe autorisé, Calculatrice interdite	

Exercice 3

Quelle est la valeur affichée dans la console après les séquences d'instructions suivantes :

```
In [23]: mins = 2
In [24]: secs = 60*mins
In [25]: mins = 5
In [26]: secs
```

120

```
In [27]: a=12
In [28]: a=5
In [29]: a=a+3
In [30]: a
```

8

```
In [31]: a=10
In [32]: b=20
In [33]: a=20
In [34]: b=a
In [35]: b
```

20

Exercice 4

```
def prix_1(poids):
    if poids > 30 :
        return 80
    elif poids > 20:
        return 50
    else:
        return 120
```

```
def prix_2(poids):
    if poids > 20 :
        return 50
    elif poids > 30:
        return 80
    else:
        return 120
```

Déterminer le résultat de l'évaluation de :

1. prix_1(10)

120

2. prix_1(40)

80

3. prix_2(25)

50

4. prix_2(45)

50

Exercice 5

Code saisi dans l'éditeur	Résultat en console
<pre>s=0 for i in range(4): s=s+1 print(s)</pre>	4
<pre>n=0 while n<15: n=n+2 print(n)</pre>	16
<pre>res = 1 for i in range(3): res=res*2 print(res)</pre>	2 4 8

Devoir Première NSI n°1	
Base Python et numération	
Première – spécialité NSI	Octobre 2020
Durée 2h	
Memo Python en annexe autorisé, Calculatrice interdite	

Exercice 6

Ecrire une fonction `aire_rectangle(la, lon)` qui retourne l'aire d'un rectangle en fonction de sa longueur et de sa largeur. Donner un exemple d'utilisation pour cette fonction.

Attention : soyez vigilant à l'indentation.

```
def aire_rectangle(la, lo):
    return la*lo
```

ou

```
def aire_rectangle(la, lo):
    print ("l'aire du rectangle est de : ", la*lo)
```

Exercice 7 (BONUS si il vous reste du temps).

Écrivez un programme qui affiche les 20 premiers termes de la table de multiplication par 75 à l'aide d'une boucle. Le résultat commencera comme ceci :

```
1 x 75 = 75
2 x 75 = 150
3 x 75 = 225
...
```

```
def multiplication(n):
    for i in range(20):
        print('{:2d}'.format(i), ' x ', n, ' = ', '{:4d}'.format(i*n))
multiplication(75)
```