

Spécialité NSI Première	DEVOIR SURVEILLE DE NSI N° 3	Vendredi 15 décembre 2023
Lycée d'Avesnières		Durée : 1 h
Année scolaire 2023-2024		Calculatrice interdite

L'énoncé complet est à rendre avec la copie.

NOM : **Prénom :**

Exercice 1 (8 points)

Pour chaque question, il y a une seule bonne réponse. Indiquer **sur la copie seulement le numéro de la question et la réponse choisie**. 1 point pour une bonne réponse, 0 point pour une mauvaise.

1) Quelle est la valeur de la variable s après l'exécution du code suivant ?

```
x = 3
y = 7//2
s = (x==y)
```

Réponses :

- 3
 - True
 - 3.5
 - False
-

2) Voici les écritures binaires de quatre nombres entiers positifs. Lequel est pair ?

Réponses :

- 10 0001
 - 10 0010
 - 11 0001
 - 11 1111
-

3) Soient P et Q deux expressions booléennes telles que P est vraie et Q est fausse.

Quelle est la valeur de l'expression (P et Q) ou (non(P) ou Q) ?

Réponses :

- Vrai
- Faux
- Ni vrai ni faux
- Vrai et faux en même temps

4) On s'intéresse à la valeur 14 présente dans la liste suivante :

```
L = [[1, 2, 3, 4, 5], [6, 7, 8, 9, 10], [11, 12, 13, 14, 15], [16, 17, 18, 19, 20]]
```

Quelle expression vaut 14 parmi les suivantes ?

Réponses :

L[2][3]

L[3][4]

L[3][2]

L[4][3]

5) Soit la liste :

```
liste = [1, [2, 3], [4, 5], 6, 7]
```

Quelle est la valeur de len(liste) ?

Réponses :

1

3

5

7

6) On considère la fonction suivante qui prend une liste de nombres en paramètre :

```
def foo(liste):  
    for i in range(len(liste)-1):  
        if liste[i] > liste[i+1]:  
            return False  
    return True
```

Quel appel de cette fonction va renvoyer True ?

Réponses :

foo([5, 4, 3, 2, 1])

foo([1, 0, 1, 0])

foo([1, 3, 2, 4])

foo([1, 2, 2, 3, 4, 5])

7) On a saisi le code suivant :

```
variable = [8, 12, -7, 52, -5, 32]
lst = [n for n in variable if n < 12]
```

Que contient la liste `lst` à la fin de l'exécution de ce script ?

Réponses :

- [8, -7, -5]
 - [8]
 - [True, False, True, False, True, False]
 - [8, 12, -7, 52, -5, 32]
-

8) Soit la suite s'instructions suivantes :

```
fruit = {}

def addone(index, dic):
    if index in dic:
        dic[index] += 1
    else:
        dic[index] = 1

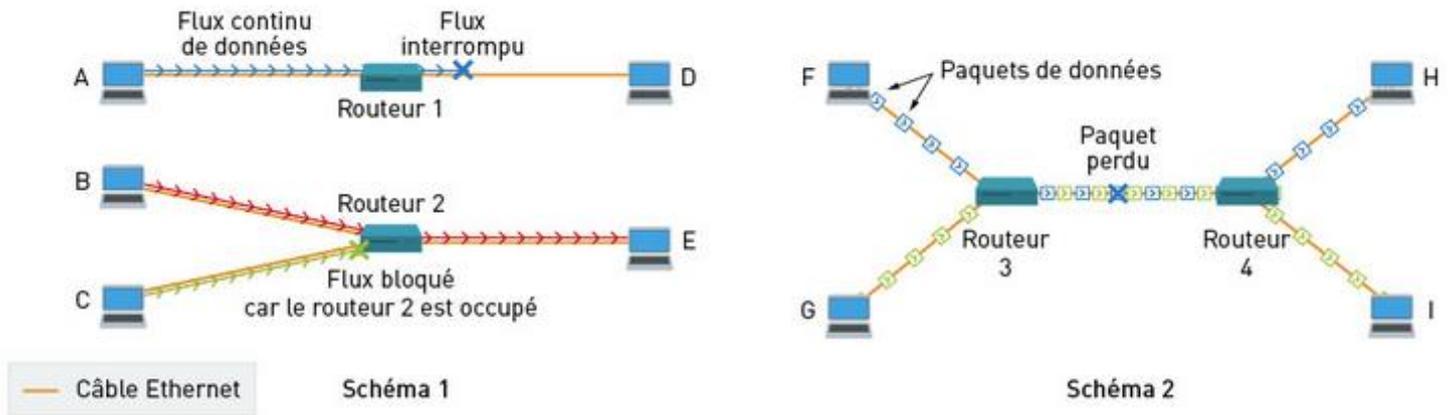
addone('Apple', fruit)
addone('Banana', fruit)
addone('Orange', fruit)
addone('Orange', fruit)
addone('Orange', fruit)
```

Que contient le dictionnaire `fruit` à la fin de l'exécution de ce script ?

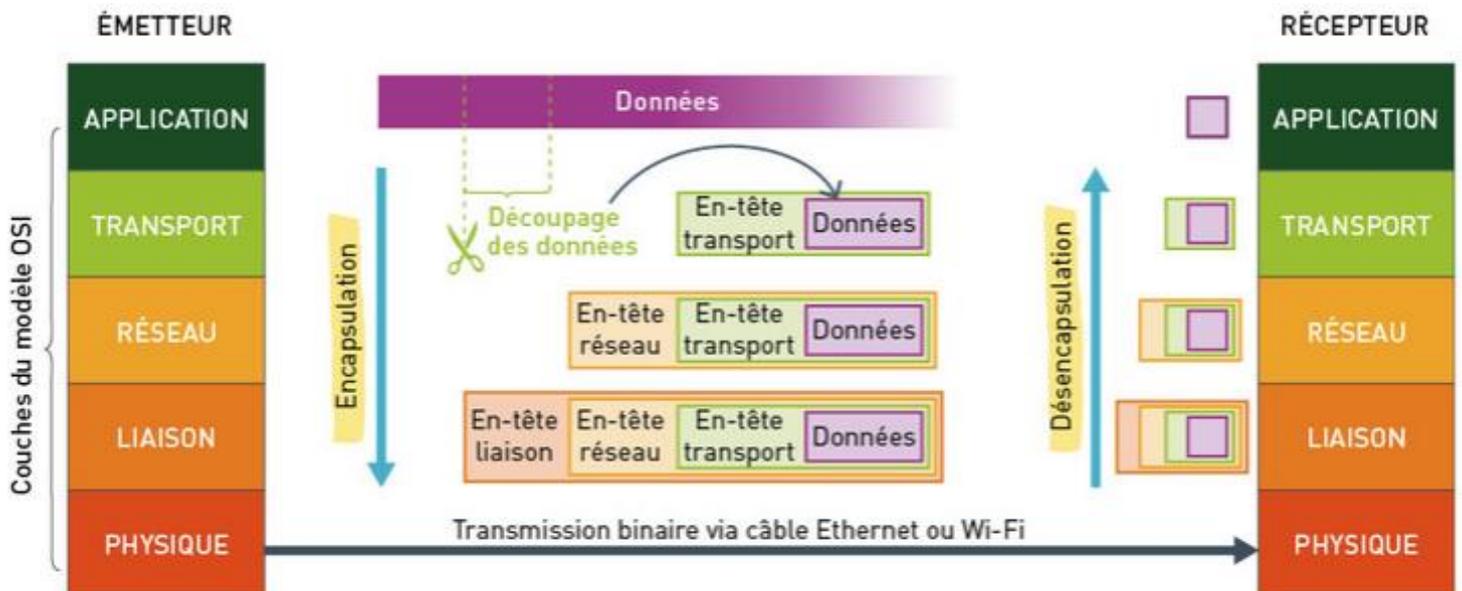
Réponses :

- {'Apple', 'Banana', 'Orange', 'Orange', 'Orange'}
 - {0: 'Apple', 1: 'Banana', 2: 'Orange'}
 - {'Apple': 1, 'Banana': 1, 'Orange': 3}
 - {1: 'Apple', 1: 'Banana', 3: 'Orange'}
-

Exercice 2 (6 points)



- 1) En observant le schéma 1, expliquer pourquoi l'envoi de données en flux continu est source d'inconvénients.
- 2) En observant le schéma 2, expliquer deux avantages du découpage des données en paquets.
- 3) En vous aidant du schéma ci-dessous, expliquer en quoi consiste le travail de la couche de transport



Exercice 3 (6 points)

Complétez directement sur l'énoncé les programmes en langage Python.

1. Remplacer une valeur

Écrire une fonction `remplacer` prenant en argument :

- une liste d'entiers `valeurs`
- un entier `valeur_cible`
- un entier `nouvelle_valeur`

et renvoyant une **nouvelle** liste contenant les mêmes valeurs que `valeurs`, dans le même ordre, sauf `valeur_cible` qui a été remplacé par `nouvelle_valeur`.

La liste passée en argument ne doit pas être modifiée.

Exemples

Si `valeurs = [3, 8, 7]`

alors `remplacer(valeurs, 3, 0)` renvoie `[0, 8, 7]` et `valeurs` vaut toujours `[3, 8, 7]`

Si `valeurs = [3, 8, 3, 5]`

alors `remplacer(valeurs, 3, 0)` renvoie `[0, 8, 0, 5]` et `valeurs` vaut toujours `[3, 8, 3, 5]`

```
def remplacer(valeurs, valeur_cible, nouvelle_valeur):
```

```
    return liste
```

2. Maximum

Écrire une fonction `maximum` prenant en argument :

- une liste non vide de nombres `nombre`

et renvoyant le plus grand élément de cette liste.

Chacun des nombres utilisés est de type `int` ou `float`. On interdit ici d'utiliser `max`, ainsi que `sort` ou `sorted`.

Exemple

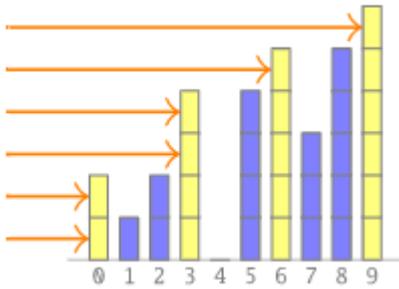
Si `nombre = [98, 12, 104, 23, 131, 9]`
alors `maximum(nombre)` renvoie `131`

```
def maximum(nombre):
```

```
    return maxi
```

3. Soleil couchant sur les bâtiments

Lorsque des bâtiments sont alignés, ils se font de l'ombre les uns les autres. Dans cet exercice, nous sommes au soleil couchant, les rayons du soleil sont donc supposés horizontaux.



Le soleil couchant éclaire neuf bâtiments, les rayons du soleil sont représentés par des flèches horizontales.

- Les bâtiments aux indices 0 et 3 reçoivent des rayons de soleil alors que les bâtiments aux indices 1 et 2 sont masqués.
- Les **quatre** bâtiments aux indices [0, 3, 6, 9] reçoivent des rayons de soleil sur au moins un étage et sont donc éclairés, alors que les autres ne le sont pas.

Écrire une fonction `nb_batiments_eclaires` qui prend en argument la liste `hauteurs` des bâtiments et qui renvoie le nombre de bâtiments éclairés.

- La hauteur des bâtiments (en nombre d'étages) est donnée par une liste d'entiers positifs. Une hauteur de zéro étage signifie l'absence de bâtiment.

Pour l'exemple ci-dessus, cette liste est [2, 1, 2, 4, 0, 4, 5, 3, 5, 6].

Exemple

Si `hauteurs = [2, 1, 2, 4, 0, 4, 5, 3, 5, 6]`
alors `nb_batiments_eclaires(hauteurs)` renvoie 4

```
def nb_batiments_eclaires(hauteurs):
```

```
    return nombre_eclaires
```