

Informatique en CPGE (2018-2019)
Devoir surveillé 3

Durée : 1 heure.

Exercice 1 (2 pts)

On utilise 5 bits pour coder les entiers relatifs.

1. Comment est codé le nombre 9 ?

2. Comment est codé le nombre -10 ?

3. Si on utilise 6 bits pour coder les entiers relatifs, combien de nombres peut-on coder et lesquels ?

Exercice 2 (4 pts)

Les flottants sont codés suivant la norme IEEE 754 sur 64 bits, soit 1 bit pour le signe, 11 bits pour l'exposant décalé et 52 bits pour la mantisse tronquée.

1. Comment est codé le nombre -4.5 ?

2. Quel est le nombre réel codé par 1011 1111 1110 1000 0000 ... 0000 ?

Exercice 3 (6 points)

1. Ecrire une fonction `multiples1` qui prend en paramètre un entier naturel non nul n et renvoie la liste des dix premiers multiples non nuls de n .

2. Ecrire une fonction `multiples2` qui prend en paramètre un entier naturel non nul `n` et renvoie la liste des multiples de `n` inférieur strictement à 1000.

3. Ecrire une fonction `produit` qui prend en paramètres une liste de nombres `nombre` et un entier naturel non nul `n` et renvoie une nouvelle liste obtenue en multipliant chaque élément de la liste `nombre` par `n`.

Exercice 4 (2 points)

Voici une fonction `mystere` :

```
def mystere(liste1, liste2):  
    liste=[]  
    i, j=0, 0  
    while i<len(liste1) and j<len(liste2):  
        if liste1[i]<liste2[j]:  
            liste.append(liste1[i])  
            i=i+1  
        else:  
            liste.append(liste2[j])  
            j=j+1  
    return liste
```

On appelle cette fonction avec l'instruction `mystere([2, 5, 6, 8], [1, 4, 7, 8, 9])`.

Quel est le résultat renvoyé ?

Exercice 5 (2 points)

Ecrire une fonction `test` qui prend en paramètre un entier naturel `n` et renvoie `True` si l'écriture de l'entier naturel `n` contient deux chiffres consécutifs identiques et `False` sinon.

Exercice 6 (2 points)

On considère la fonction suivante :

```
def prog(n):  
    s1=n[-1]  
    s2=int(n[-1])  
    for i in range(1,len(n)):  
        s1=s1+n[-1-i]  
        s2=s2+int(n[-1-i])  
    return s1,s2
```

Que renvoie la fonction avec l'appel `prog("8617")` ?

Exercice 7 (2 points)

On donne les programmes `prog0` et `prog1` suivants où n est un entier naturel.

```
1 def prog0(n):  
2     if n==1 :  
3         return False  
4     if n==2:  
5         return True  
6     for d in range(2,n):  
7         if n%d==0:  
8             return False  
9     return True
```

```
1 def prog1(n):  
2     if n==1 :  
3         return False  
4     if n==2:  
5         return True  
6     for d in range(2,n):  
7         if n%d==0:  
8             return False  
9     return True
```

Que renvoient les appels `prog0(7)`, `prog0(15)`, `prog1(7)`, `prog1(15)` ?