

## Interro – Récursivité – Corrigé

### Exercice 1 (3 points)

a. `mystere([1,9,11,42],2)` renvoie :

```

return True and mystere([1,9,11,42],1) car t[1]<t[2]
      |
      return True and mystere([1,9,11,42],0)
            |
            return True

```

On fait donc True and True and True : c'est True.

b. Si on a un False, la fonction renverra False. La fonction `mystere` détermine si le tableau est dans l'ordre croissant de 0 à i.

### Exercice 2 (4 points)

a.  $Ack(0,1) = 1 + 1 = 2$        $Ack(0,2) = 1 + 2 = 3$        $Ack(1,0) = Ack(0,1) = 2$   
 $Ack(2,0) = Ack(1,1) = Ack(0, Ack(1,0)) = Ack(0,2) = 3$   
 $Ack(1,2) = Ack(0, Ack(1,1)) = Ack(0,3) = 4$

b. Attention aux else, elif et autres conditions. À l'écrit et en l'absence d'ordinateur, else/if (sans elif) est une valeur sure...

```

def Ack(m, n):
    assert m>=0 and n>=0, 'm et n ne
    peuvent pas être négatifs'
    if m==0: return n+1
    elif n==0: return Ack(m-1,1)
    elif n>0: return Ack(m-1, Ack(m, n-1))

```

### Exercice 3 (2 points)

Je vous réfère au cours du vendredi  
 13 novembre pour la réponse à la question a.

b. L'erreur est que l'on traite les cases rouges (déjà colorées) comme des cases blanches (à colorier) . S'ensuit, un petit problème de récursion infinie (on monte, on colorie, on descend, on colorie, on descend, mur, on monte, on colorie etc.)

Pour régler le problème, il suffit de changer la ligne 2 en :

```
if t[pos_i][pos_j]!="0":
```