

Défi 1- Proposition de solutions

Solution 1 Gestion des bits de poids faible et des bits de poids fort : Comme $2^4 = 16$ alors la division à quotient entier de a par 16 donne :

$$a = 16b + c$$

b est le quotient et c est le reste.

```
c=a%16
```

Solution 2

```
def cacher(imag1,imag2,imag3):
    """ introduit imag1
    dans imag2 (meme taille)
    pour donner imag3
    """
    f=im.imread(imag1)
    g=im.imread(imag2)
    t1=np.shape(f)
    t2=np.shape(g)
    assert t1[0]==t2[0] and t1[1]==t2[1], 'ENC'
    for i in range(t1[0]):
        for j in range(t1[1]):
            for k in range(3):
                g[i,j,k]=(g[i,j,k]//16)*16
                +f[i,j,k]//16
    im.imsave(imag3,g)
```

```
def trouver(imag1,imag2):
    """ Cherche dans imag1
    l'image sur les 4 LSB
    reconstitue imag2, l'image cachee
    """
    f=im.imread(imag1)
    t=np.shape(f)
    for i in range(t[0]):
        for j in range(t[1]):
            for k in range(3):
                f[i,j,k]=(f[i,j,k]%16)*16
    im.imsave(imag2,f)
```

Pour relever le défi, il convient de taper :

```
trouver('defi.tif','image.tif')
```

