

**Informatique PCSI**  
**Corrigé TP 7 : algorithmes (manipulations  
d'une image)**

**Exercice 1**

```
def inverser_nb(img):
    haut, larg = len(img), len(img[0])
    for i in range(haut):
        for j in range(larg):
            img[i][j] = 1 - img[i][j]
```

**Exercice 2**

```
def sym_vert(img):
    haut, larg = len(img), len(img[0])
    for i in range(haut):
        for j in range(larg//2):
            img[i][j], img[i][larg-1-j] = img[i][larg-1-j], img[i][j]
```

```
def sym_hor(img):
    haut, larg = len(img), len(img[0])
    for i in range(haut//2):
        img[i], img[haut-1-i] = img[haut-1-i], img[i]
```

**Exercice 3**

```
def rotation(img, x, y, n):
    if n > 1:
        n = n // 2
        for i in range(x, x+n):
            for j in range(y, y+n):
                temp = img[i][j]
                img[i][j] = img[i][n + j]
                img[i][n + j] = img[n + i][n + j]
                img[n + i][n + j] = img[n + i][j]
                img[n + i][j] = temp
        rotation(img, x, y, n)
        rotation(img, x, y+n, n)
        rotation(img, x+n, y, n)
        rotation(img, x+n, y+n, n)
```

**Exercice 4**

```

def reduction(img, d):
    n, p = len(img), len(img[0])
    mat1 = [[0 for j in range(p)] for i in range(n//d + 1)]
    for i in range(n):
        if i % d == 0:
            for j in range(p):
                mat1[i//d][j] = img[i][j]
    mat2 = [[0 for j in range(p//d + 1)] for i in range(n//d + 1)]
    for i in range(n//d + 1):
        for j in range(p):
            if j % d == 0:
                mat2[i][j//d] = mat1[i][j]
    return mat2

```

```

def agrandissement(img, d):
    n, p = len(img), len(img[0])
    mat1 = [[0 for j in range(d*p)] for i in range(n)]
    for i in range(n):
        for j in range(p):
            if img[i][j] == 1:
                for k in range(d):
                    mat1[i][d*j+k] = 1
    mat2 = [[0 for j in range(d*p)] for i in range(d*n)]
    for i in range(n):
        for j in range(d*p):
            if mat1[i][j] == 1:
                for k in range(d):
                    mat2[d*i+k][j] = 1
    return mat2

```