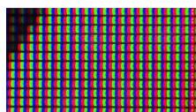
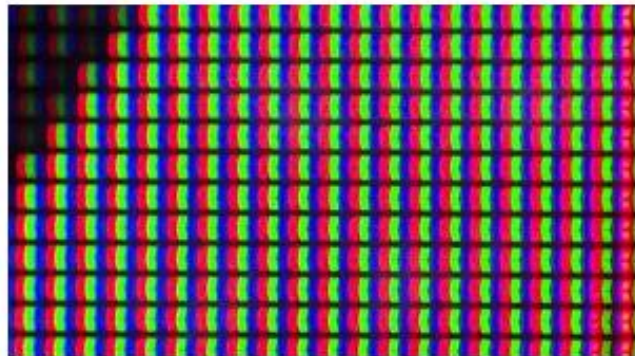
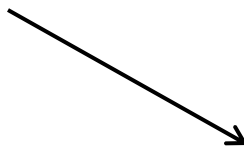
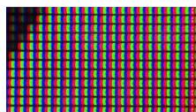


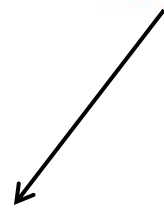
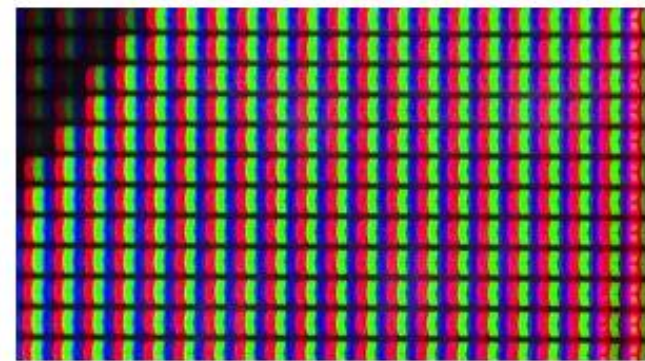
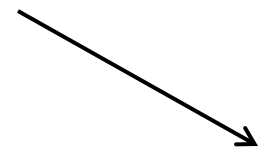
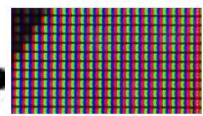
Présentation de la notion de liste en informatique.

Application aux images numériques.

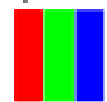
1. Les images numériques.
2. Les listes.
3. Les listes de listes.
4. Des calculs.
5. Autres applications.

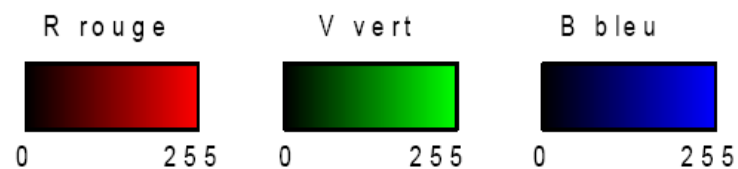
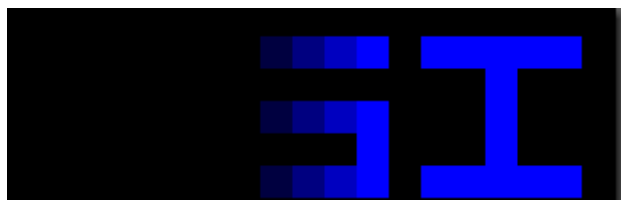






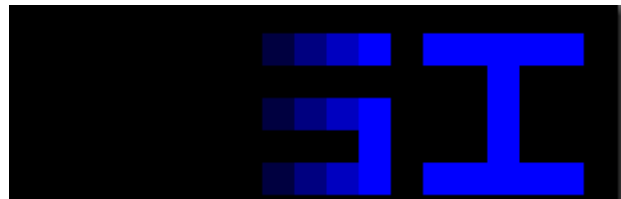
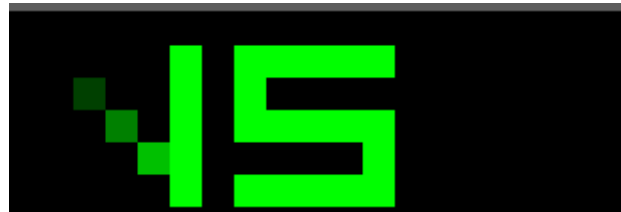
1 pixel



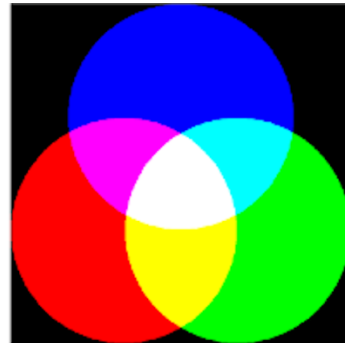


Intensité de 0 à 255

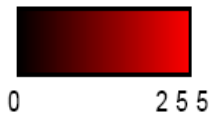
Codage 3 x 8 bits ou 3 octets / pixel



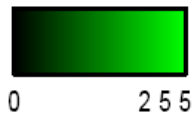
Synthèse additive



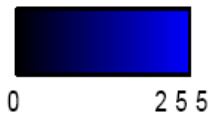
R rouge



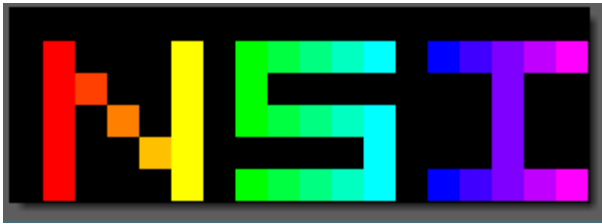
V vert



B bleu



Intensité de 0 à 255

[illegible][illegible]

Questionnaire à
Choix
Multiples 1

Pour chaque question, donner l'unique bonne réponse



1. Dans une image numérique, quelles sont les trois composantes couleurs ?

a. noir, blanc, gris

b. rouge, vert, bleu

c. chihuahua, muffin, chat

d. rouge, orange, jaune



1. Dans une image numérique, quelles sont les trois composantes couleurs ?

a. noir, blanc, gris

b. rouge, vert, bleu

c. chihuahua, muffin, chat

d. rouge, orange, jaune

- 
2. Chaque composante couleur d'un pixel est codée avec des nombres entiers entre 0 et 255. Combien de couleurs peut-on coder ?
- a. 256
 - b. 3
 - c. $768 = 3 \times 256$
 - d. $256 \times 256 \times 256 = 16\,777\,216$



2. Chaque composante couleur d'un pixel est codée avec des nombres entiers entre 0 et 255. Combien de couleurs peut-on coder ?

a. 256

b. 3

c. $768 = 3 \times 256$

d. $256 \times 256 \times 256 = 16\,777\,216$

- 
3. Un pixel couleur est codé par le triplet (255,255,255). Quelle est la couleur de ce pixel ?
- a. noir
 - b. gris
 - c. blanc
 - d. ce pixel ne peut pas exister



3. Un pixel couleur est codé par le triplet (255,255,255). Quelle est la couleur de ce pixel ?
- a. noir
 - b. gris
 - c. blanc**
 - d. ce pixel ne peut pas exister

Codage 1 bit/px



0

1

NSI

[illegible]

[illegible]

[illegible]

[illegible]

y\x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
2	0	85	106	0	0	170	0	85	0	0	0	0	0	0	0	127	0	0	0
3	0	85	0	127	0	170	0	85	106	127	149	170	0	0	0	127	0	0	0
4	0	85	0	0	149	170	0	0	0	0	0	170	0	0	0	127	0	0	0
5	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

y\x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
2	0	85	106	0	0	170	0	85	0	0	0	0	0	0	0	127	0	0	0
3	0	85	0	127	0	170	0	85	106	127	149	170	0	0	0	127	0	0	0
4	0	85	0	0	149	170	0	0	0	0	0	170	0	0	0	127	0	0	0
5	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

y\x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
2	0	85	106	0	0	170	0	85	0	0	0	0	0	0	0	127	0	0	0
3	0	85	0	127	0	170	0	85	106	127	149	170	0	0	0	127	0	0	0
4	0	85	0	0	149	170	0	0	0	0	0	170	0	0	0	127	0	0	0
5	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

[0,85,0,0,0,170,0,85,106,127,149,170,0,85,106,149,170,0]

y\x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
2	0	85	106	0	0	170	0	85	0	0	0	0	0	0	0	127	0	0	0
3	0	85	0	127	0	170	0	85	106	127	149	170	0	0	0	127	0	0	0
4	0	85	0	0	149	170	0	0	0	0	0	170	0	0	0	127	0	0	0
5	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

[0,85,0,0,0,170,0,85,106,127,149,170,0,85,106,149,170,0]

...

...

[0,85,0,0,0,170,0,85,106,127,149,170,0,85,106,149,170,0]

[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

Comment créer une liste ?

Comment créer une liste ?

Combien d'éléments dans une liste ?

Comment créer une liste ?

Combien d'éléments dans une liste ?

Comment accéder à une valeur ?

Comment créer une liste ?

Combien d'éléments dans une liste ?

Comment accéder à une valeur ?

Comment modifier une valeur ?

Comment créer une liste ?

Combien d'éléments dans une liste ?

Comment accéder à une valeur ?

Comment modifier une valeur ?

Comment parcourir une liste ?

Questionnaire à
Choix
Multiples 2

Pour chaque question, donner l'unique bonne réponse



On donne la liste $L = [1, 5, 8, 2, 3]$

1. Que renvoie l'instruction `len(L)` ?

a. 5

b. 19

c. 1

d. 4



On donne la liste $L = [1, 5, 8, 2, 3]$

1. Que renvoie l'instruction `len(L)` ?

a. 5

b. 19

c. 1

d. 4



On donne la liste $\mathbb{L} = [1, 5, 8, 2, 3]$

2. Que renvoie l'instruction $\mathbb{L}[2]$?

a. 3

b. 4

c. 5

d. 8



On donne la liste $\mathbb{L} = [1, 5, 8, 2, 3]$

2. Que renvoie l'instruction $\mathbb{L}[2]$?

a. 3

b. 4

c. 5

d. 8



On donne la liste $\mathbb{L} = [1, 5, 8, 2, 3]$

3. Que renvoie l'instruction $\mathbb{L}[5]$?

- a. rien
- b. 3
- c. une erreur
- d. chihuahua



On donne la liste $\mathbb{L} = [1, 5, 8, 2, 3]$

3. Que renvoie l'instruction $\mathbb{L}[5]$?

a. rien

b. 3

c. une erreur

d. chihuahua

y\x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
2	0	85	106	0	0	170	0	85	0	0	0	0	0	0	0	127	0	0	0
3	0	85	0	127	0	170	0	85	106	127	149	170	0	0	0	127	0	0	0
4	0	85	0	0	149	170	0	0	0	0	0	170	0	0	0	127	0	0	0
5	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

[0,85,0,0,0,170,0,85,106,127,149,170,0,85,106,149,170,0]

...

...

[0,85,0,0,0,170,0,85,106,127,149,170,0,85,106,149,170,0]

[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

y\x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
2	0	85	106	0	0	170	0	85	0	0	0	0	0	0	0	127	0	0	0
3	0	85	0	127	0	170	0	85	106	127	149	170	0	0	0	127	0	0	0
4	0	85	0	0	149	170	0	0	0	0	0	170	0	0	0	127	0	0	0
5	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

[0,85,0,0,0,170,0,85,106,127,149,170,0,85,106,149,170,0]

...

...

[0,85,0,0,0,170,0,85,106,127,149,170,0,85,106,149,170,0]

[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

img = [[], [], , [], []]

$$L = [[12 , 248 , 56] , [235 , 165 , 28]]$$

$L = [[12 , 248 , 56] , [235 , 165 , 28]]$

Comment accéder à une valeur ?

$L = [[12 , 248 , 56] , [235 , 165 , 28]]$

Comment accéder à une valeur ?

Quelle est la valeur de $L[0][2]$?

$L = [[12 , 248 , 56] , [235 , 165 , 28]]$

Comment accéder à une valeur ?

Quelle est la valeur de $L[0][2]$?

Comment modifier une valeur ?

$L = [[12 , 248 , 56] , [235 , 165 , 28]]$

Comment accéder à une valeur ?

Quelle est la valeur de $L[0][2]$?

Comment modifier une valeur ?

Comment parcourir la liste de listes ?

Questionnaire à
Choix
Multiples 3

Pour chaque question, donner l'unique bonne réponse



On donne la liste

$L = [[100, 15, 18, 203], [21, 24, 134, 79], [50, 27, 24, 19]]$

1. Que renvoie l'instruction $L[1]$?

a. 100

b. $[21, 24, 134, 79]$

c. $[100, 15, 18, 203]$

d. 21



On donne la liste

$L = [[100, 15, 18, 203], [21, 24, 134, 79], [50, 27, 24, 19]]$

1. Que renvoie l'instruction $L[1]$?

a. 100

b. [21, 24, 134, 79]

c. [100, 15, 18, 203]

d. 21



On donne la liste

$L = [[100, 15, 18, 203], [21, 24, 134, 79], [50, 27, 24, 19]]$

2. Que renvoie l'instruction $L[2][3]$?

a. une erreur

b. 134

c. 19

d. 27



On donne la liste

$L = [[100, 15, 18, 203], [21, 24, 134, 79], [50, 27, 24, 19]]$

2. Que renvoie l'instruction $L[2][3]$?

a. une erreur

b. 134

c. 19

d. 27



On donne la liste

$L = [[100, 15, 18, 203], [21, 24, 134, 79], [50, 27, 24, 19]]$

3. Que renvoie l'instruction `len(L)` ?

a. une erreur

b. 3

c. 4

d. 12



On donne la liste

$L = [[100, 15, 18, 203], [21, 24, 134, 79], [50, 27, 24, 19]]$

3. Que renvoie l'instruction `len(L)` ?

a. une erreur

b. 3

c. 4

d. 12



On donne la liste

$L = [[100, 15, 18, 203], [21, 24, 134, 79], [50, 27, 24, 19]]$

4. Que renvoie l'instruction `len(L[0])` ?

a. une erreur

b. 3

c. 4

d. 12



On donne la liste

$L = [[100, 15, 18, 203], [21, 24, 134, 79], [50, 27, 24, 19]]$

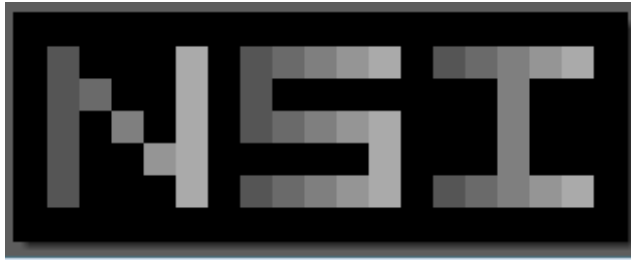
4. Que renvoie l'instruction `len(L[0])` ?

a. une erreur

b. 3

c. 4

d. 12



```
img = [  
[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  
[0,85,0,0,0,170,0,85,106,127,149,170,0,85,106,127,149,170,0],  
[0,85,106,0,0,170,0,85,0,0,0,0,0,0,0,127,0,0,0],  
[0,85,0,127,0,170,0,85,106,127,149,170,0,0,0,127,0,0,0],  
[0,85,0,0,149,170,0,0,0,0,0,170,0,0,0,127,0,0,0],  
[0,85,0,0,0,170,0,85,106,127,149,170,0,85,106,127,149,170,0],  
[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]  
]
```



SeuilNB(img, seuil) avec seuil = 150



```
new = [  
[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],  
[0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,1,0],  
[0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0],  
[0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,0],  
[0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,0],  
[0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,1,0],  
[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]  
]
```



Faire des calculs

```
1 def SeuilNB(img, seuil) :
2     ''' Produit une image en noir et blanc à partir de la matrice
3         img d'une image en niveaux de gris.
4         seuil, entier compris entre 0 et 255, indique la valeur à
5         partir de laquelle les pixels deviennent blancs ou noirs.
6     '''
7     hauteur = len(img) # nombre de lignes (hauteur de l'image)
8     largeur = len(img[0]) # nombre de colonnes (largeur de l'image)
9     new = [[0 for x in range(largeur)] for y in range(hauteur)]
10    for y in range(hauteur) :
11        for x in range(largeur) :
12            if img[y][x] >= seuil :
13                new[y][x] = 1 # Le pixel est blanc
14            else :
15                new[y][x] = 0 # Le pixel est noir
16    return new
```



Robert Doisneau - L'information scolaire - 1956



Seuil 50



Seuil 70



Seuil 100



Seuil 127



Seuil 150



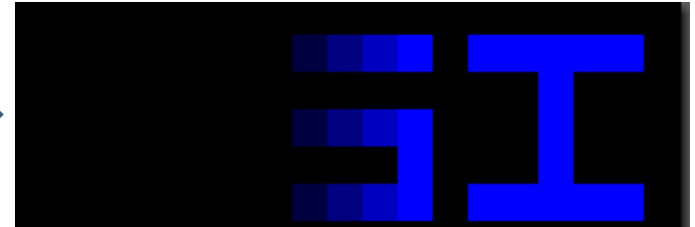
yx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)
1	(0,0,0)	(255,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(255,255,0)	(0,0,0)	(0,255,0)	(0,255,64)	(0,255,128)	(0,255,192)	(0,255,255)	(0,0,0)	(0,0,255)	(64,0,255)	(128,0,255)	(192,0,255)	(255,0,255)	(0,0,0)
2	(0,0,0)	(255,0,0)	(255,64,0)	(0,0,0)	(0,0,0)	(255,255,0)	(0,0,0)	(0,255,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(128,0,255)	(0,0,0)	(0,0,0)	(0,0,0)
3	(0,0,0)	(255,0,0)	(0,0,0)	(255,128,0)	(0,0,0)	(255,255,0)	(0,0,0)	(0,255,0)	(0,255,64)	(0,255,128)	(0,255,192)	(0,255,255)	(0,0,0)	(0,0,0)	(0,0,0)	(128,0,255)	(0,0,0)	(0,0,0)	(0,0,0)
4	(0,0,0)	(255,0,0)	(0,0,0)	(0,0,0)	(255,192,0)	(255,255,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,255,255)	(0,0,0)	(0,0,0)	(0,0,0)	(128,0,255)	(0,0,0)	(0,0,0)	(0,0,0)
5	(0,0,0)	(255,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(255,255,0)	(0,0,0)	(0,255,0)	(0,255,64)	(0,255,128)	(0,255,192)	(0,255,255)	(0,0,0)	(0,0,255)	(64,0,255)	(128,0,255)	(192,0,255)	(255,0,255)	(0,0,0)
6	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)



Filtre bleu(img)

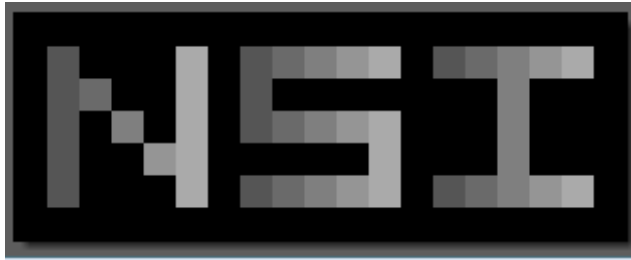


yx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)
1	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,64)	(0,0,128)	(0,0,192)	(0,0,255)	(0,0,0)	(0,0,255)	(0,0,255)	(0,0,255)	(0,0,255)	(0,0,255)	(0,0,0)
2	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,255)	(0,0,0)	(0,0,0)	(0,0,0)
3	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,64)	(0,0,128)	(0,0,255)	(0,0,255)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,255)	(0,0,0)	(0,0,0)	(0,0,0)
4	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,255)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,255)	(0,0,0)	(0,0,0)	(0,0,0)
5	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,64)	(0,0,128)	(0,0,255)	(0,0,255)	(0,0,0)	(0,0,255)	(0,0,255)	(0,0,255)	(0,0,255)	(0,0,255)	(0,0,0)
6	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)



Faire des calculs

```
1 def FiltreBleu(img) :  
2     ''' Produit une image bleue à partir d'une image en  
3         couleur donnée sous forme matricielle (liste de lignes)  
4         en mettant les intensités de rouge et de vert à 0.  
5     '''  
6     hauteur = len(img) # nombre de lignes (hauteur de l'image)  
7     largeur = len(img[0]) # nombre de colonnes (largeur de l'image)  
8     new = [[0 for x in range(largeur)] for y in range(hauteur)]  
9     for y in range(hauteur) :  
10        for x in range(largeur) :  
11            new[y][x] = [0,0,img[y][x][2]]  
12    return new
```

y\x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
2	0	85	106	0	0	170	0	85	0	0	0	0	0	0	0	127	0	0	0
3	0	85	0	127	0	170	0	85	106	127	149	170	0	0	0	127	0	0	0
4	0	85	0	0	149	170	0	0	0	0	0	170	0	0	0	127	0	0	0
5	0	85	0	0	0	170	0	85	106	127	149	170	0	85	106	127	149	170	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



FiltreGrisNegatif(img)



y\x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
1	255	170	255	255	255	85	255	170	149	128	106	85	255	170	149	128	149	85	255
2	255	170	149	255	255	85	255	170	255	255	255	255	255	255	128	255	255	255	255
3	255	170	255	128	255	85	255	170	149	128	106	85	255	255	255	128	255	255	255
4	255	170	255	255	106	85	255	255	255	255	255	85	255	255	255	128	255	255	255
5	255	170	255	255	255	85	255	170	149	128	106	85	255	170	149	128	106	85	255
6	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255



Faire des calculs

```
1 def FiltreGrisNegatif(img) :  
2     hauteur = len(img) # nombre de lignes (hauteur de l'image)  
3     largeur = len(img[0]) # nombre de colonnes (largeur de l'image)  
4     new = [[0 for x in range(largeur)] for y in range(hauteur)]  
5     for y in range(hauteur) :  
6         for x in range(largeur) :  
7             new[y][x] = 255-img[y][x]  
8     return new
```

Faire des calculs

```
1 def FiltreGrisNegatif(img) :
2     hauteur = len(img) # nombre de lignes (hauteur de l'image)
3     largeur = len(img[0]) # nombre de colonnes (largeur de l'image)
4     new = [[0 for x in range(largeur)] for y in range(hauteur)]
5     for y in range(hauteur) :
6         for x in range(largeur) :
7             new[y][x] = 255-img[y][x]
8     return new
```

```
1 def FiltreCouleurNegatif(img) :
2     hauteur = len(img) # nombre de lignes (hauteur de l'image)
3     largeur = len(img[0]) # nombre de colonnes (largeur de l'image)
4     new = [[0 for x in range(largeur)] for y in range(hauteur)]
5     for y in range(hauteur) :
6         for x in range(largeur) :
7             pix = img[y][x]
8             new[y][x] = [255-pix[0],255-pix[1],255-pix[2]]
9     return new
```



y\x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)
1	(0,0,0)	(255,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(255,255,0)	(0,0,0)	(0,255,0)	(0,255,64)	(0,255,128)	(0,255,192)	(0,255,255)	(0,0,0)	(0,0,255)	(64,0,255)	(128,0,255)	(192,0,255)	(255,0,255)	(0,0,0)
2	(0,0,0)	(255,0,0)	(255,64,0)	(0,0,0)	(0,0,0)	(255,255,0)	(0,0,0)	(0,255,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(128,0,255)	(0,0,0)	(0,0,0)	(0,0,0)
3	(0,0,0)	(255,0,0)	(0,0,0)	(255,128,0)	(0,0,0)	(255,255,0)	(0,0,0)	(0,255,0)	(0,255,64)	(0,255,128)	(0,255,192)	(0,255,255)	(0,0,0)	(0,0,0)	(0,0,0)	(128,0,255)	(0,0,0)	(0,0,0)	(0,0,0)
4	(0,0,0)	(255,0,0)	(0,0,0)	(0,0,0)	(255,192,0)	(255,255,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,255,255)	(0,0,0)	(0,0,0)	(0,0,0)	(128,0,255)	(0,0,0)	(0,0,0)	(0,0,0)
5	(0,0,0)	(255,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(255,255,0)	(0,0,0)	(0,255,0)	(0,255,64)	(0,255,128)	(0,255,192)	(0,255,255)	(0,0,0)	(0,0,255)	(64,0,255)	(128,0,255)	(192,0,255)	(255,0,255)	(0,0,0)
6	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)

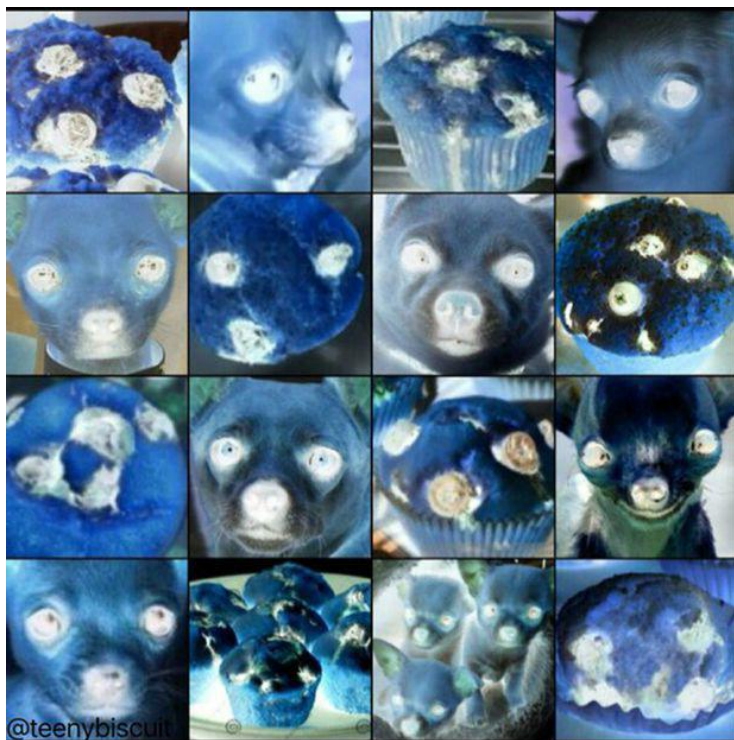


FiltreCouleurNegatif(img)



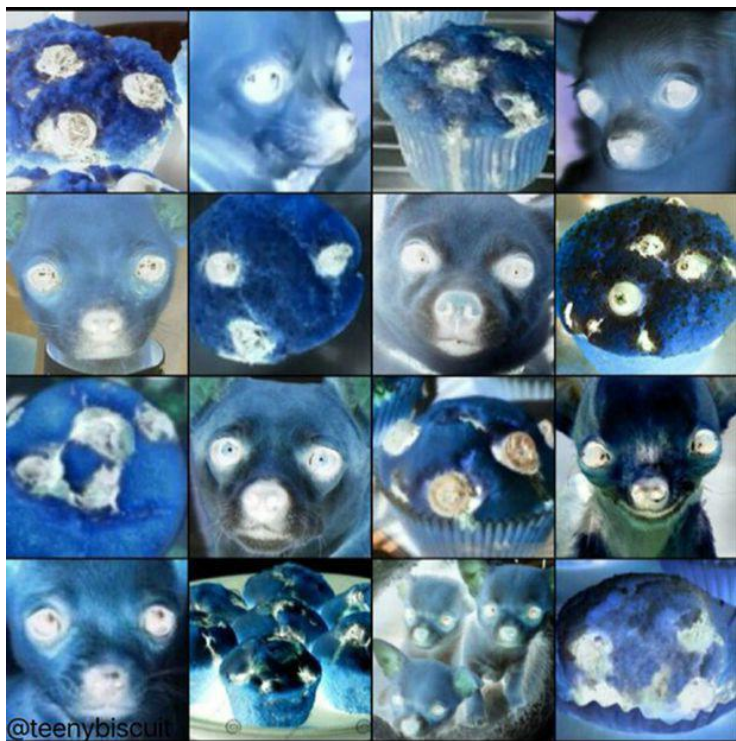
y\x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)
1	(255,255,255)	(0,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(0,0,255)	(255,255,255)	(255,0,255)	(255,0,191)	(255,0,127)	(255,0,63)	(255,0,0)	(255,255,255)	(255,255,0)	(191,255,0)	(127,255,0)	(63,255,0)	(0,255,0)	(255,255,255)
2	(255,255,255)	(0,255,255)	(0,191,255)	(255,255,255)	(255,255,255)	(0,0,255)	(255,255,255)	(255,0,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(127,255,0)	(255,255,255)	(255,255,255)	(255,255,255)
3	(255,255,255)	(0,255,255)	(255,255,255)	(0,127,255)	(255,255,255)	(0,0,255)	(255,255,255)	(255,0,255)	(255,0,191)	(255,0,127)	(255,0,63)	(255,0,0)	(255,255,255)	(255,255,255)	(255,255,255)	(127,255,0)	(255,255,255)	(255,255,255)	(255,255,255)
4	(255,255,255)	(0,255,255)	(255,255,255)	(255,255,255)	(0,0,255)	(0,0,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,0,0)	(255,255,255)	(255,255,255)	(255,255,255)	(127,255,0)	(255,255,255)	(255,255,255)	(255,255,255)
5	(255,255,255)	(0,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(0,0,255)	(255,255,255)	(255,0,255)	(255,0,191)	(255,0,127)	(255,0,63)	(255,0,0)	(255,255,255)	(255,255,255)	(255,255,0)	(191,255,0)	(127,255,0)	(63,255,0)	(0,255,0)
6	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)	(255,255,255)





Voici le négatif d'une image en couleur.
Que représente l'image de départ ?

- a. des fraises
- b. des étoiles
- c. des otaries
- d. des myrtilles



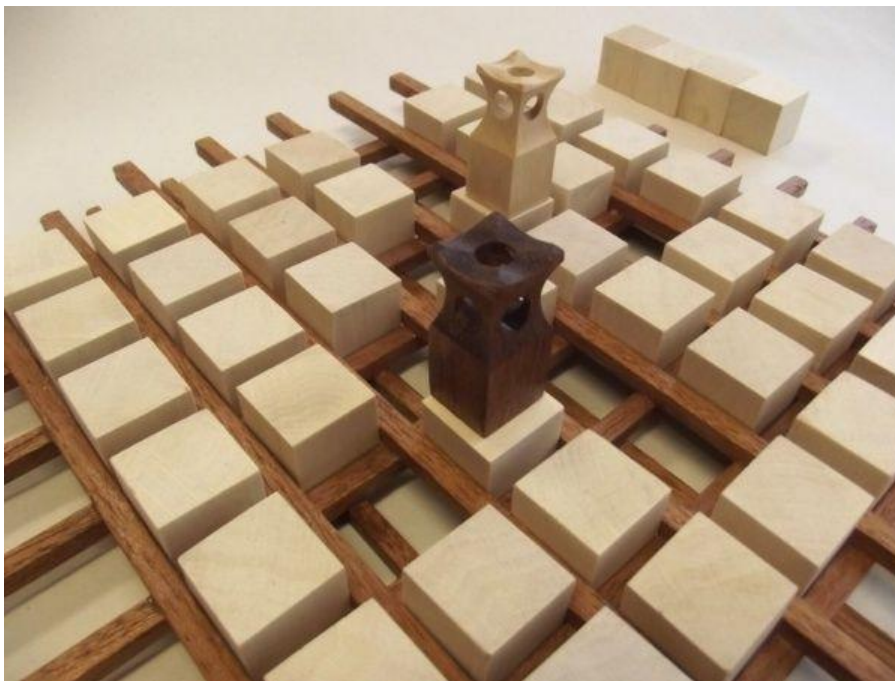


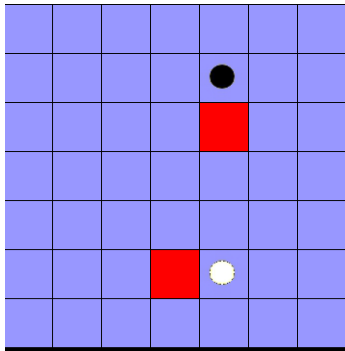
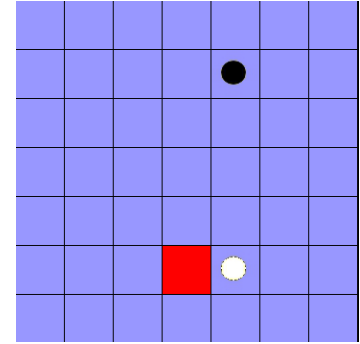
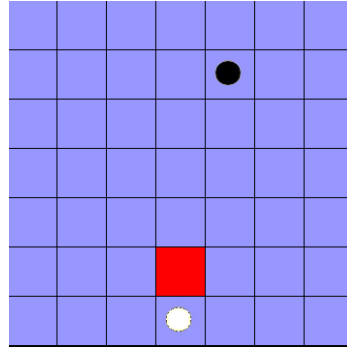
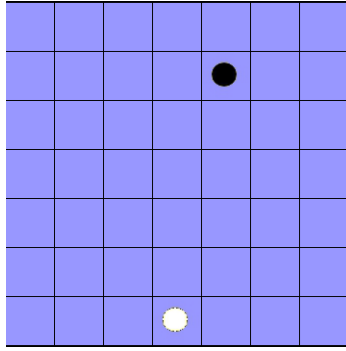
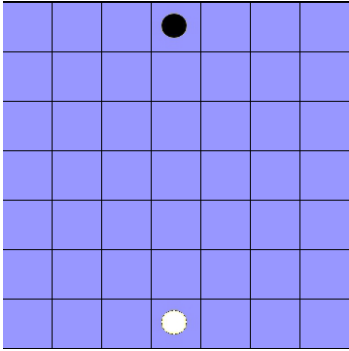
Dames



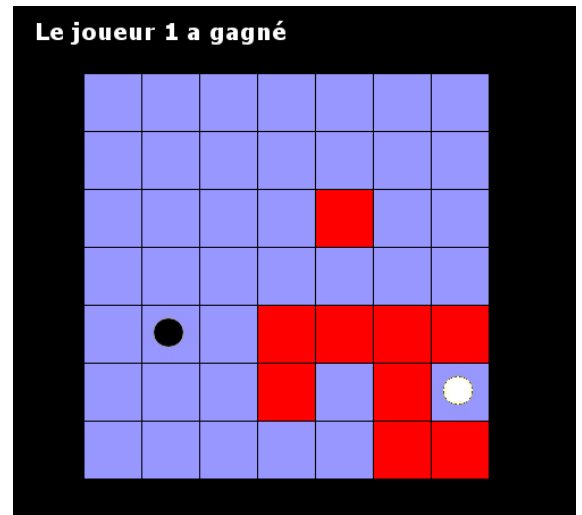
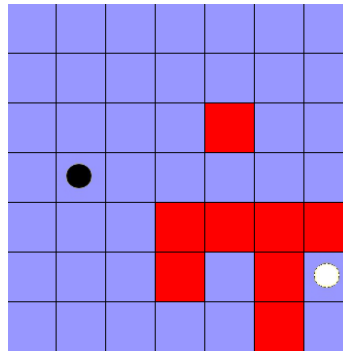
Puisse 4

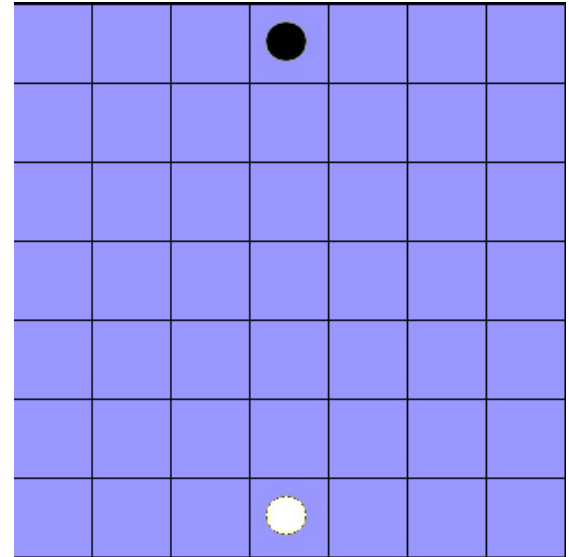
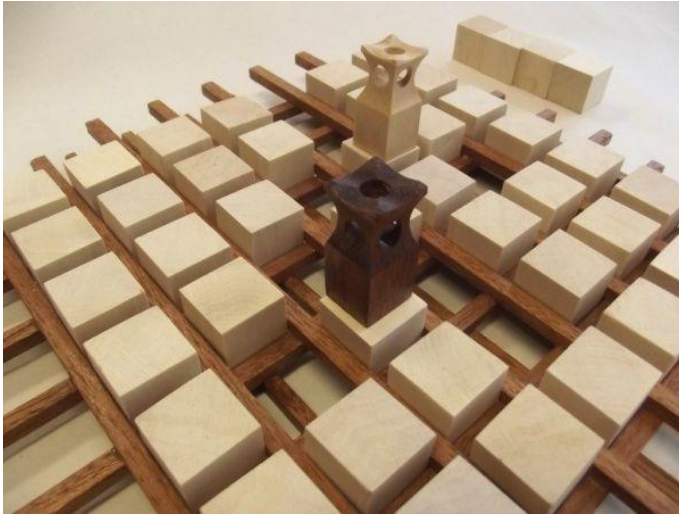
ISOLA





...





ISOLA

```
jeu = [[-1,-1,-1,-1,-1,-1,-1,-1,-1],
        [-1,2,2,2,0,2,2,2,-1],
        [-1,2,2,2,2,2,2,2,-1],
        [-1,2,2,2,2,2,2,2,-1],
        [-1,2,2,2,2,2,2,2,-1],
        [-1,2,2,2,2,2,2,2,-1],
        [-1,2,2,2,2,2,2,2,-1],
        [-1,2,2,2,1,2,2,2,-1],
        [-1,-1,-1,-1,-1,-1,-1,-1,-1]]
```