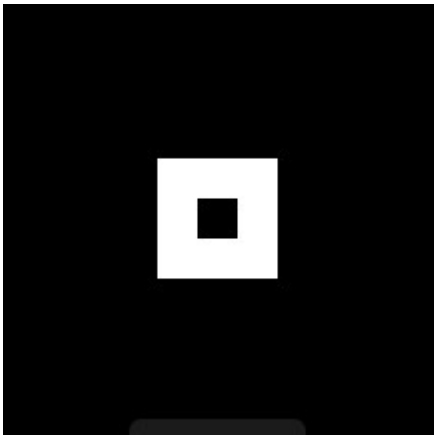
Compte-rendu Partie 1 Tache 3

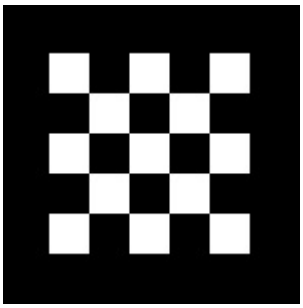
Dans cette partie du compte rendu, nous allons vous présenter le résultat d'évaluation du contour par notre algorithme.

Nous allons utiliser un programme nommé test_contour qui nous permet de tester l'évaluation d'un unique contour.

Toutes les images que nous avons créées pour tester ceci se situent dans le fichier Images_tests_perso.

| Nom | Image PBM | Evaluation du premier contour |
|---|---|---|
| Cube.pbm |  | contour detecter !<br>nombre de points : 5<br>nombre de segment : 4<br>Point (1.000000, 2.000000)<br>Point (2.000000, 2.000000)<br>Point (2.000000, 1.000000)<br>Point (1.000000, 1.000000)<br>Point (1.000000, 2.000000)<br><br>fin des points du contours |

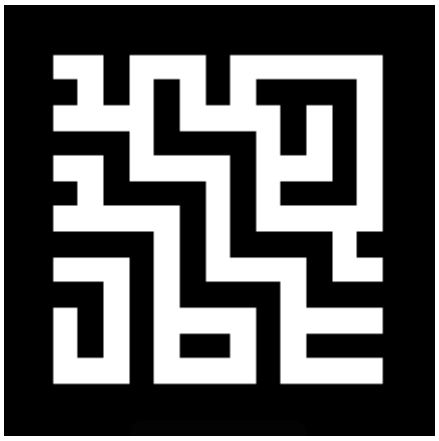
Damier.pbm



```
raybac38@raybac38-Z68P-DS3:~/D
Images_tests_perso/damier.pbm
contour detecter !
nombre de points : 29
nombre de segment : 28
Point (1.000000, 5.000000)
Point (2.000000, 5.000000)
Point (2.000000, 4.000000)
Point (3.000000, 4.000000)
Point (3.000000, 5.000000)
Point (4.000000, 5.000000)
Point (4.000000, 4.000000)
Point (5.000000, 4.000000)
Point (5.000000, 3.000000)
Point (4.000000, 3.000000)
Point (4.000000, 2.000000)
Point (5.000000, 2.000000)
Point (5.000000, 1.000000)
Point (4.000000, 1.000000)
Point (4.000000, 0.000000)
Point (3.000000, 0.000000)
Point (3.000000, 1.000000)
Point (2.000000, 1.000000)
Point (2.000000, 0.000000)
Point (1.000000, 0.000000)
Point (1.000000, 1.000000)
Point (0.000000, 1.000000)
Point (0.000000, 2.000000)
Point (1.000000, 2.000000)
Point (1.000000, 3.000000)
Point (0.000000, 3.000000)
Point (0.000000, 4.000000)
Point (1.000000, 4.000000)
Point (1.000000, 5.000000)

fin des points du contours
```

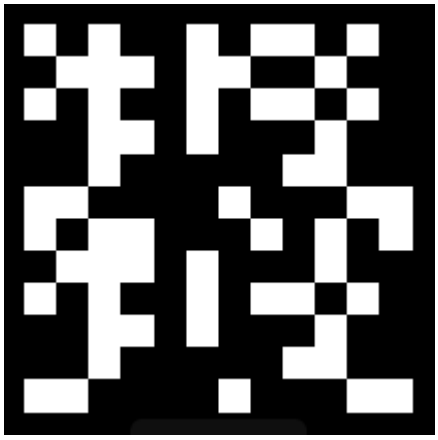
Laby.pbm

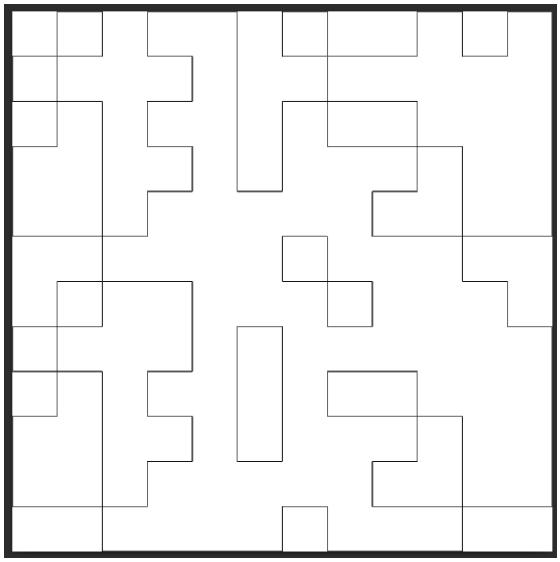


```
Images_tests_perso/laby.pbm
contour detecter !
nombre de points : 61
nombre de segment : 60
Point (0.000000, 15.000000)
Point (1.000000, 15.000000)
Point (2.000000, 15.000000)
Point (3.000000, 15.000000)
Point (4.000000, 15.000000)
Point (5.000000, 15.000000)
Point (6.000000, 15.000000)
Point (7.000000, 15.000000)
Point (8.000000, 15.000000)
Point (9.000000, 15.000000)
Point (10.000000, 15.000000)
Point (11.000000, 15.000000)
Point (12.000000, 15.000000)
Point (13.000000, 15.000000)
Point (14.000000, 15.000000)
Point (15.000000, 15.000000)
Point (15.000000, 14.000000)
Point (15.000000, 13.000000)
Point (15.000000, 12.000000)
Point (15.000000, 11.000000)
Point (15.000000, 10.000000)
Point (15.000000, 9.000000)
Point (15.000000, 8.000000)
Point (15.000000, 7.000000)
Point (15.000000, 6.000000)
Point (15.000000, 5.000000)
Point (15.000000, 4.000000)
Point (15.000000, 3.000000)
Point (15.000000, 2.000000)
Point (15.000000, 1.000000)
Point (15.000000, 0.000000)
Point (14.000000, 0.000000)
Point (13.000000, 0.000000)
Point (12.000000, 0.000000)
Point (11.000000, 0.000000)
Point (10.000000, 0.000000)
Point (9.000000, 0.000000)
Point (8.000000, 0.000000)
Point (7.000000, 0.000000)
Point (6.000000, 0.000000)
Point (5.000000, 0.000000)
Point (4.000000, 0.000000)
Point (3.000000, 0.000000)
Point (2.000000, 0.000000)
Point (1.000000, 0.000000)
Point (0.000000, 0.000000)
Point (0.000000, 1.000000)
Point (0.000000, 2.000000)
Point (0.000000, 3.000000)
Point (0.000000, 4.000000)
Point (0.000000, 5.000000)
Point (0.000000, 6.000000)
Point (0.000000, 7.000000)
Point (0.000000, 8.000000)
Point (0.000000, 9.000000)
Point (0.000000, 10.000000)
Point (0.000000, 11.000000)
Point (0.000000, 12.000000)
Point (0.000000, 13.000000)
Point (0.000000, 14.000000)
Point (0.000000, 15.000000)

fin des points du contours
```

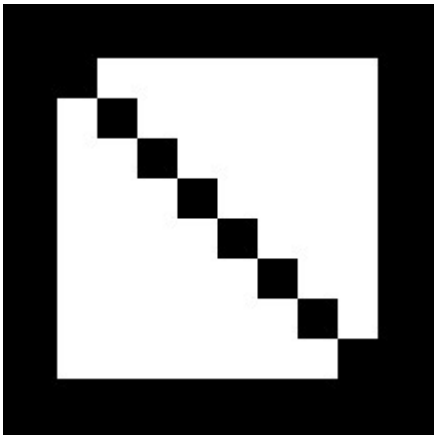
Random.pbm





Nous vous proposons un affichage

Diagonal.
pbm



graphique car il y a un trop grand nombre de points.

```
contour detecter !
nombre de points : 33
nombre de segment : 32
Point (0.000000, 8.000000)
Point (1.000000, 8.000000)
Point (1.000000, 7.000000)
Point (2.000000, 7.000000)
Point (2.000000, 6.000000)
Point (3.000000, 6.000000)
Point (3.000000, 5.000000)
Point (4.000000, 5.000000)
Point (4.000000, 4.000000)
Point (5.000000, 4.000000)
Point (5.000000, 3.000000)
Point (6.000000, 3.000000)
Point (6.000000, 2.000000)
Point (7.000000, 2.000000)
Point (7.000000, 1.000000)
Point (8.000000, 1.000000)
Point (8.000000, 0.000000)
Point (7.000000, 0.000000)
Point (7.000000, 1.000000)
Point (6.000000, 1.000000)
Point (6.000000, 2.000000)
Point (5.000000, 2.000000)
Point (5.000000, 3.000000)
Point (4.000000, 3.000000)
Point (4.000000, 4.000000)
Point (3.000000, 4.000000)
Point (3.000000, 5.000000)
Point (2.000000, 5.000000)
Point (2.000000, 6.000000)
Point (1.000000, 6.000000)
Point (1.000000, 7.000000)
Point (0.000000, 7.000000)
Point (0.000000, 8.000000)

fin des points du contours
```