Carres_Magiques1

February 22, 2023

1 Carrés magiques normaux

Marc Lorenzi

18 février 2023

[1]:
```python
import matplotlib.pyplot as plt
import random
import math
```

1.1 1. Introduction

1.1.1 1.1 Qu'est-ce qu'une matrice magique ?

Dans toute la suite, nous supposerons que les indices des coefficients d'une matrice commencent à 0. Toutes les matrices considérées seront carrées. Nous appellerons *taille* d'une matrice (carrée) le nombre de ses lignes (et donc aussi de ses colonnes).

Définition 1. Soit $n \geq 1$. Une *matrice magique de taille* n est une matrice $C \in \mathcal{M}_n(\mathbb{R})$ dont les sommes des coefficients sur chaque ligne, chaque colonne, et chacune des deux diagonales sont identiques. En d'autres termes, C est magique si et seulement si il existe $S \in \mathbb{R}$ tel que

- Pour tout $i \in [|0, n-1|]$, $\sum_{j=0}^{n-1} C_{ij} = S$.
- Pour tout $j \in [|0, n-1|]$, $\sum_{i=0}^{n-1} C_{ij} = S$.
- $\sum_{i=0}^{n-1} C_{ii} = S$.
- $\sum_{i=0}^{n-1} C_{i(n-1-i)} = S$.

Nous noterons $S = \sigma(C)$. Le réel $\sigma(C)$ est la *somme magique* de C. Nous noterons également M_n l'ensemble des matrices magiques de taille n.

Définition 2. Un *carré* est une matrice dont les coefficients sont des entiers naturels non nuls.

Définition 3. Un *carré normal* de taille $n \geq 1$ est un carré dont les coefficients sont les entiers 1, 2,..., n^2.

Le but de ce notebook est, principalement, de décrire un algorithme permettant, pour tout entier $n \geq 3$ de déterminer un carré magique normal de taille n.

1.1.2 1.2 La somme d'un carré magique normal

Soit C un carré magique normal de taille n. On a

$$\sum_{i,j=0}^{n-1} C_{ij} = \sum_{k=1}^{n^2} k = \frac{1}{2}n^2(n^2+1)$$

Par ailleurs,

$$\sum_{i,j=0}^{n-1} C_{ij} = \sum_{i=0}^{n-1}\left(\sum_{j=0}^{n-1} C_{ij}\right) = n\sigma(C)$$

Ainsi,

$$\sigma(C) = \frac{1}{2}n(n^2+1)$$

[2]:
```
def somme_magique(n):
    return n * (n ** 2 + 1) // 2
```

Voici la valeur des sommes magiques des carrés magiques pour les premières valeurs de n.

[3]:
```
for n in range(1, 21):
    print('%2d%5d' % (n, somme_magique(n)))
```

```
 1    1
 2    5
 3   15
 4   34
 5   65
 6  111
 7  175
 8  260
 9  369
10  505
11  671
12  870
13 1105
14 1379
15 1695
16 2056
17 2465
18 2925
19 3439
20 4010
```

[4]:
```
print(somme_magique(2023))
```

```
4139594095
```

Proposition. Il existe un unique carré magique normal de taille 1.

Démonstration. C'est (1).

Proposition. Il n'existe pas de carré magique normal de taille 2.

Démonstration. Soit $C = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$. Supposons que C est un carré magique normal. On a alors $\sigma(C) = 5$. L'un des coefficients de C est égal à 1. Par exemple, $a = 1$. On a alors $a + c = a + b = a + d = 5$ et donc $b = c = d = 4$, contradiction.

Proposition. Soit $n \geq 3$. Il existe un carré magique normal de taille n.

Démonstration. C'est plus ou moins le but de ce notebook. Patience …

1.1.3 1.3 Quelques fonctions utiles

Nous représenterons une matrice $A \in \mathcal{M}_n(\mathbb{R})$ en Python par une liste (que nous noterons encore A) de n listes de longueur n. Pour tous $i, j \in [|0, n-1|]$, décidons que le coefficient A_{ij} de la matrice A est $A[i][j]$. $A[i]$ est donc la $i$ème ligne de la matrice A.

La taille d'un matrice est la longueur de la liste qui la représente.

[5]:
```
def taille(A): return len(A)
```

La fonction `matrice_vide` renvoie une matrice de taille n dont tous les coefficients sont égaux à `None`.

[6]:
```
def matrice_vide(n):
    A = n * [None]
    for i in range(n):
        A[i] = n * [None]
    return A
```

La fonction `somme_ligne` renvoie la somme des coefficients de la ligne i de la matrice A.

[7]:
```
def somme_ligne(A, i):
    n = taille(A)
    s = 0
    for j in range(n):
        s += A[i][j]
    return s
```

La fonction `somme_colonne` renvoie la somme des coefficients de la colonne j de la matrice A.

[8]:
```
def somme_colonne(A, j):
    n = taille(A)
```

```python
    s = 0
    for i in range(n):
        s += A[i][j]
    return s
```

La fonction somme_diag1 renvoie la somme des coefficients de la diagonale « principale » de la matrice A.

[9]:
```python
def somme_diag1(A):
    n = taille(A)
    s = 0
    for i in range(n): s += A[i][i]
    return s
```

La fonction somme_diag2 renvoie la somme des coefficients de la diagonale « secondaire » de la matrice A.

[10]:
```python
def somme_diag2(A):
    n = taille(A)
    s = 0
    for i in range(n): s += A[i][n - 1 - i]
    return s
```

La fonction est_matrice_magique renvoie True si la matrice A est magique et False sinon.

[11]:
```python
def est_matrice_magique(A):
    s = somme_ligne(A, 0)
    n = taille(A)
    for i in range(n):
        if somme_ligne(A, i) != s: return False
        if somme_colonne(A, i) != s: return False
    if somme_diag1(A) != s: return False
    if somme_diag2(A) != s: return False
    return True
```

La fonction est_normal renvoie True si les coefficients de la matrice A de taille n sont les entiers entre 1 et n^2, et False sinon.

[12]:
```python
def est_normal(A):
    n = taille(A)
    s = [A[i][j] for i in range(n) for j in range(n)]
    s.sort()
    for k in range(n ** 2):
        if s[k] != k + 1: return False
    return True
```

La fonction est_carre_magique_normal renvoie True si C est un carré magique normal et False sinon.

[13]:
```
def est_carre_magique_normal(C):
    return est_matrice_magique(C) and est_normal(C)
```

La fonction `print_carre` afiche joliment un carré. Évidemment, on n'y voit pas grand chose si la taille du carré est trop grande.

[14]:
```
def print_carre(C):
    n = taille(C)
    for i in range(n):
        print('+' + n * '----+')
        print('|', end='')
        for j in range(n):

            print('%4d|' % C[i][j], end='')
        print()
    print('+' + n * '----+')
```

La fonction `plot_matrice` affiche une représentation graphique de la matrice C.

[15]:
```
def plot_matrice(C):
    plt.imshow(C, cmap='hot', interpolation='none')
```

1.1.4 1.3 Trois exemples

Voici un carré magique normal de taille 3. Nous verrons plus loin que, à rotations et symétries près, c'est le seul possible.

[16]:
```
carre3 = [[4, 9, 2], [3, 5, 7], [8, 1, 6]]
```

[17]:
```
print_carre(carre3)
```

```
+----+----+----+
|   4|   9|   2|
+----+----+----+
|   3|   5|   7|
+----+----+----+
|   8|   1|   6|
+----+----+----+
```

[18]:
```
print(est_carre_magique_normal(carre3))
```

```
True
```

Voici un carré magique normal de taille 4. On part de la matrice de taille 4 contenant les entiers de 1 à 16 rangés dans l'ordre.

[19]:
```
carre4 = [[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12], [13, 14, 15, 16]]
```

[20]:
```
print_carre(carre4)
```

```
+----+----+----+----+
|   1|   2|   3|   4|
+----+----+----+----+
|   5|   6|   7|   8|
+----+----+----+----+
|   9|  10|  11|  12|
+----+----+----+----+
|  13|  14|  15|  16|
+----+----+----+----+
```

On échange ensuite les coins opposés.

[21]:
```
carre4[0][0], carre4[3][3] = carre4[3][3], carre4[0][0]
carre4[0][3], carre4[3][0] = carre4[3][0], carre4[0][3]
print_carre(carre4)
```

```
+----+----+----+----+
|  16|   2|   3|  13|
+----+----+----+----+
|   5|   6|   7|   8|
+----+----+----+----+
|   9|  10|  11|  12|
+----+----+----+----+
|   4|  14|  15|   1|
+----+----+----+----+
```

On échange enfin les coins opposés du carré central 2×2.

[22]:
```
carre4[1][1], carre4[2][2] = carre4[2][2], carre4[1][1]
carre4[1][2], carre4[2][1] = carre4[2][1], carre4[1][2]
print_carre(carre4)
```

```
+----+----+----+----+
|  16|   2|   3|  13|
+----+----+----+----+
|   5|  11|  10|   8|
+----+----+----+----+
|   9|   7|   6|  12|
+----+----+----+----+
|   4|  14|  15|   1|
+----+----+----+----+
```

Ce carré est un carré magique normal.

[23]:
```
print(est_carre_magique_normal(carre4))
```

```
True
```

Un carré magique normal apparaît dans le tableau d'Albrecht Dürer *Melencolia*. On laisse au lecteur le soin de le vérifier.

Voici enfin un carré magique normal de taille 8. Comment l'a-t-on obtenu ? Attendez la fin du notebook …

[24]:
```
carre8 = [
  [33,  13,  30,  21,   6,  64,  40,  53],
  [49,   8,  35,  27,  17,  36,  62,  26],
  [46,  55,  31,  38,  42,   1,   2,  45],
  [43,  60,  54,  51,   4,   3,  11,  34],
  [14,  15,  16,  47,  59,  39,   9,  61],
```

```
 [37,  32,  25,  41,  52,  10,  58,   5],
 [18,  48,  19,  28,  23,  44,  56,  24],
 [20,  29,  50,   7,  57,  63,  22,  12]]
```

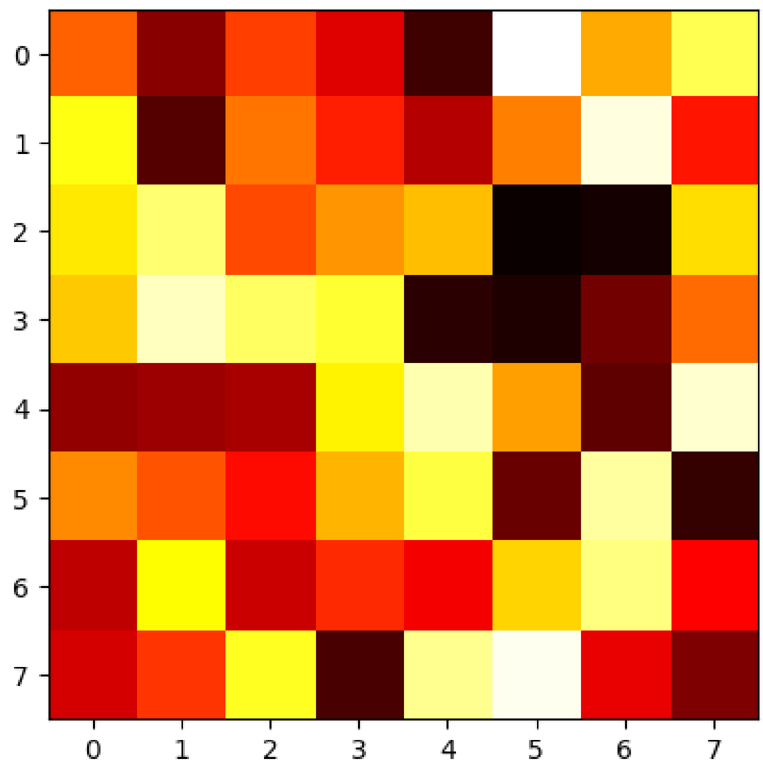
[25]:
```python
print_carre(carre8)
```

```
+----+----+----+----+----+----+----+----+
|  33|  13|  30|  21|   6|  64|  40|  53|
+----+----+----+----+----+----+----+----+
|  49|   8|  35|  27|  17|  36|  62|  26|
+----+----+----+----+----+----+----+----+
|  46|  55|  31|  38|  42|   1|   2|  45|
+----+----+----+----+----+----+----+----+
|  43|  60|  54|  51|   4|   3|  11|  34|
+----+----+----+----+----+----+----+----+
|  14|  15|  16|  47|  59|  39|   9|  61|
+----+----+----+----+----+----+----+----+
|  37|  32|  25|  41|  52|  10|  58|   5|
+----+----+----+----+----+----+----+----+
|  18|  48|  19|  28|  23|  44|  56|  24|
+----+----+----+----+----+----+----+----+
|  20|  29|  50|   7|  57|  63|  22|  12|
+----+----+----+----+----+----+----+----+
```

[26]:
```python
print(est_carre_magique_normal(carre8))
```

```
True
```

[27]:
```python
plot_matrice(carre8)
```



Nous aurons besoin plus tard des carrés magiques normaux de tailles 4 et 8 pour fabriquer des carrés magiques normaux de taille quelconque.

1.1.5 1.4 Rotations et symétries

Le **groupe dihédral** $\mathbb{D}_4$ est l'ensemble des isométries du plan laissant invariant l'ensemble $\{(-1,-1),(-1,1),(1,-1),(1,1)\}$ (c'est l'ensemble des sommets d'un carré). $\mathbb{D}_4$ contient 8 éléments :

$$\mathbb{D}_4 = \{id, r, r^2, r^3, s, r \circ s, r^2 \circ s, r^3 \circ s\}$$

où r est la rotation de centre O, d'angle $\frac{\pi}{2}$ et s est la symétrie orthogonale par rapport à Ox. $\mathbb{D}_4$ est un groupe pour la composition des applications.

Soit $n \geq 1$. Le groupe $\mathbb{D}_4$ *agit* sur $\mathcal{M}_n(\mathbb{R})$, en ce sens que l'on peut définir une multiplication externe $\times : \mathbb{D}_4 \times \mathcal{M}_n(\mathbb{R}) \longrightarrow \mathcal{M}_n(\mathbb{R})$. Il suffit de définir, pour tout $A \in \mathcal{M}_n(\mathbb{R})$, rA et sA, les autres produits s'en déduisant par associativité. Par exemple $(r^2 \circ s)A = r(r(sA))$.

Définissons rA et sA en posant, pour tous $i, j \in [|0, n-1|]$,

$$(rA)_{ij} = A_{j(n-1-i)}$$

$$(sA)_{ij} = A_{(n-1-i)j}$$

De façon plus imagée, rA est la matrice obtenue en faisant subir à A un quart de tour dans le sens antihoraire, et sA est la matrice obtenue en échangeant les lignes 0 et $n-1$, les lignes 1 et $n-2$, etc., de la matrice A.

Remarque. Pour tous $i, j \in [|0, n-1|]$,

$$((s \circ r)A)_{ij} = (s(rA))_{ij} = (rA)_{(n-1-i)j} = A_{j(n-1-(n-1-i)} = A_{ji}$$

Ainsi, $(s \circ r)A = A^T$ est la transposée de A.

[28]:
```
def rotation(A):
    n = taille(A)
    B = matrice_vide(n)
    for i in range(n):
        for j in range(n):
            B[i][j] = A[j][n - 1 - i]
    return B
```

[29]:
```
A = carre8
B = rotation(A)
print_carre(A)
```

```
+----+----+----+----+----+----+----+----+
|  33|  13|  30|  21|   6|  64|  40|  53|
+----+----+----+----+----+----+----+----+
|  49|   8|  35|  27|  17|  36|  62|  26|
+----+----+----+----+----+----+----+----+
|  46|  55|  31|  38|  42|   1|   2|  45|
+----+----+----+----+----+----+----+----+
|  43|  60|  54|  51|   4|   3|  11|  34|
+----+----+----+----+----+----+----+----+
|  14|  15|  16|  47|  59|  39|   9|  61|
+----+----+----+----+----+----+----+----+
|  37|  32|  25|  41|  52|  10|  58|   5|
+----+----+----+----+----+----+----+----+
|  18|  48|  19|  28|  23|  44|  56|  24|
+----+----+----+----+----+----+----+----+
|  20|  29|  50|   7|  57|  63|  22|  12|
+----+----+----+----+----+----+----+----+
```

[30]:
```
print_carre(B)
```

```
+----+----+----+----+----+----+----+----+
|  53|  26|  45|  34|  61|   5|  24|  12|
+----+----+----+----+----+----+----+----+
```

```
|  40|  62|   2|  11|   9|  58|  56|  22|
+----+----+----+----+----+----+----+----+
|  64|  36|   1|   3|  39|  10|  44|  63|
+----+----+----+----+----+----+----+----+
|   6|  17|  42|   4|  59|  52|  23|  57|
+----+----+----+----+----+----+----+----+
|  21|  27|  38|  51|  47|  41|  28|   7|
+----+----+----+----+----+----+----+----+
|  30|  35|  31|  54|  16|  25|  19|  50|
+----+----+----+----+----+----+----+----+
|  13|   8|  55|  60|  15|  32|  48|  29|
+----+----+----+----+----+----+----+----+
|  33|  49|  46|  43|  14|  37|  18|  20|
+----+----+----+----+----+----+----+----+
```

[31]:

```python
def symetrie(A):
    n = taille(A)
    B = matrice_vide(n)
    for i in range(n):
        for j in range(n):
            B[i][j] = A[n - 1 - i][j]
    return B
```

[32]:

```python
A = carre8
B = symetrie(A)
print_carre(A)
```

```
+----+----+----+----+----+----+----+----+
|  33|  13|  30|  21|   6|  64|  40|  53|
+----+----+----+----+----+----+----+----+
|  49|   8|  35|  27|  17|  36|  62|  26|
+----+----+----+----+----+----+----+----+
|  46|  55|  31|  38|  42|   1|   2|  45|
+----+----+----+----+----+----+----+----+
|  43|  60|  54|  51|   4|   3|  11|  34|
+----+----+----+----+----+----+----+----+
|  14|  15|  16|  47|  59|  39|   9|  61|
+----+----+----+----+----+----+----+----+
|  37|  32|  25|  41|  52|  10|  58|   5|
+----+----+----+----+----+----+----+----+
|  18|  48|  19|  28|  23|  44|  56|  24|
+----+----+----+----+----+----+----+----+
|  20|  29|  50|   7|  57|  63|  22|  12|
+----+----+----+----+----+----+----+----+
```

[33]:

```python
print_carre(B)
```

```
+----+----+----+----+----+----+----+----+
```

```
|  20|  29|  50|   7|  57|  63|  22|  12|
+----+----+----+----+----+----+----+----+
|  18|  48|  19|  28|  23|  44|  56|  24|
+----+----+----+----+----+----+----+----+
|  37|  32|  25|  41|  52|  10|  58|   5|
+----+----+----+----+----+----+----+----+
|  14|  15|  16|  47|  59|  39|   9|  61|
+----+----+----+----+----+----+----+----+
|  43|  60|  54|  51|   4|   3|  11|  34|
+----+----+----+----+----+----+----+----+
|  46|  55|  31|  38|  42|   1|   2|  45|
+----+----+----+----+----+----+----+----+
|  49|   8|  35|  27|  17|  36|  62|  26|
+----+----+----+----+----+----+----+----+
|  33|  13|  30|  21|   6|  64|  40|  53|
+----+----+----+----+----+----+----+----+
```

Proposition. Soit A un carré magique normal. Pour tout $f \in \mathbb{D}_4$, fA est un carré magique normal.

Démonstration. Il suffit de faire la vérification pour $f = r$ et $f = s$. Faisons-le pour r. Posons $B = rA$. La matrice B a les mêmes coefficients que A, elle est donc normale. Remarquons que les lignes de B sont des colonnes de A, les colonnes de B sont des lignes de A « renversées », et les diagonales de B sont des diagonales (éventuellement renversées) de A. Il en résulte facilement que B est magique.

Remarque. À partir d'un carré magique normal A de taille différente de 1, on peut donc créer 7 autres carrés magiques normaux. Voici par exemple la transposée de notre carré magique de taille 8.

[34]:
```
A = carre8
B = symetrie(rotation(A))
print(est_carre_magique_normal(B))
print_carre(B)
```

```
True
+----+----+----+----+----+----+----+----+
|  33|  49|  46|  43|  14|  37|  18|  20|
+----+----+----+----+----+----+----+----+
|  13|   8|  55|  60|  15|  32|  48|  29|
+----+----+----+----+----+----+----+----+
|  30|  35|  31|  54|  16|  25|  19|  50|
+----+----+----+----+----+----+----+----+
|  21|  27|  38|  51|  47|  41|  28|   7|
+----+----+----+----+----+----+----+----+
|   6|  17|  42|   4|  59|  52|  23|  57|
+----+----+----+----+----+----+----+----+
|  64|  36|   1|   3|  39|  10|  44|  63|
+----+----+----+----+----+----+----+----+
```

```
|  40|  62|   2|  11|   9|  58|  56|  22|
+----+----+----+----+----+----+----+----+
|  53|  26|  45|  34|  61|   5|  24|  12|
+----+----+----+----+----+----+----+----+
```

1.2 2. Les carrés magiques normaux de taille 3

Soit $n \geq 1$. L'ensemble M_n est un sous-espace vectoriel de $\mathcal{M}_n(\mathbb{R})$: pour toutes matrices magiques $C, C' \in M_n$ et tout réel λ, $C + C' \in M_n$ et $\lambda C \in M_n$. De plus, la fonction « somme magique » $\sigma : M_n \longrightarrow M_n$ est une application linéaire :

- $\sigma(C + C') = \sigma(C) + \sigma(C')$.
- $\sigma(\lambda C) = \lambda\sigma(C)$.

Ce résultat se prouve sans difficulté.

1.2.1 2.1 Matrices magiques symétriques et antisymétriques

Notons $\mathcal{S}$ l'ensemble des matrices magiques 3×3 symétriques et $\mathcal{A}$ l'ensemble des matrices magiques 3×3 antisymétriques.

Proposition. Les ensembles $\mathcal{S}$ et $\mathcal{A}$ sont des sous-espaces vectoriels de M_3. On a de plus $\mathcal{S} \oplus \mathcal{A} = M_3$.

Démonstration. Si $A \in M_3$, la transposée de A, A^T, appartient à M_3. La transposition étant linéaire, l'application $\varphi : A \longmapsto A^T$ est donc un endomorphisme de M_3. De plus, $\varphi^2 = id$, donc φ est une symétrie de M_3. De là, $\mathcal{A} = \ker(\varphi - id)$ et $\mathcal{S} = \ker(\varphi + id)$ sont deux sous-espaces vectoriels supplémentaires de M_3.

Pour déterminer les matrices magiques de taille 3, il suffit donc de déterminer celles qui sont symétriques et celles qui sont antisymétriques, puis de les ajouter.

1.2.2 2.2 Matrices magiques antisymétriques

Une matrice antisymétrique a ses coefficients diagonaux nuls. On a donc pour tout $A \in \mathcal{A}$, $\sigma(A) = 0$. Écrivons

$$A = \begin{pmatrix} 0 & a & b \\ -a & 0 & c \\ -b & -c & 0 \end{pmatrix}$$

On a $A \in \mathcal{A}$ si et seulement si

$$\begin{cases} a + b & = & 0 \\ -a + c & = & 0 \\ -b - c & = & 0 \end{cases}$$

ce qui équivaut à $b = -a$, $c = a$. Les matrices de $\mathcal{A}$ sont donc les matrices de la forme $A = \lambda W$, où $\lambda \in \mathbb{R}$ et

$$W = \begin{pmatrix} 0 & 1 & -1 \\ -1 & 0 & 1 \\ 1 & -1 & 0 \end{pmatrix}$$

Ainsi, $\mathcal{A} =< W >$ est de dimension 1.

1.2.3 2.2 Matrices magiques symétriques

Déterminons tout d'abord les matrices $A \in \mathcal{S}$ telles que $\sigma(A) = 0$. Écrivons

$$A = \begin{pmatrix} a & b & c \\ b & d & e \\ c & e & f \end{pmatrix}$$

On a $A \in \mathcal{S}$ et $\sigma(A) = 0$ si et seulement si

$$(S) \left\{ \begin{array}{lcl} a+b+c & = & 0 \\ b+d+e & = & 0 \\ c+e+f & = & 0 \\ a+d+f & = & 0 \\ 2c+d & = & 0 \end{array} \right.$$

Résolvons (S) par l'algorithme du pivot de Gauss.

Étape 1 : $L_4 \leftarrow L_4 - L_1$.

$$(S) \left\{ \begin{array}{lcl} a+b+c & = & 0 \\ b+d+e & = & 0 \\ c+e+f & = & 0 \\ -b-c+d+f & = & 0 \\ 2c+d & = & 0 \end{array} \right.$$

Étape 2 : $L_1 \leftarrow L_1 - L_2$, $L_4 \leftarrow L_4 + L_2$.

$$(S) \left\{ \begin{array}{lcl} a+c-d-e & = & 0 \\ b+d+e & = & 0 \\ c+e+f & = & 0 \\ -c+2d+e+f & = & 0 \\ 2c+d & = & 0 \end{array} \right.$$

Étape 3 : $L_1 \leftarrow L_1 - L_3$, $L_4 \leftarrow \frac{1}{2}(L_4 + L_3)$, $L_5 \leftarrow L_5 - 2L_3$.

$$(S)\ \begin{cases} a-d-2e-f &= 0 \\ b+d+e &= 0 \\ c+e+f &= 0 \\ d+e+f &= 0 \\ d-2e-2f &= 0 \end{cases}$$

Étape 4 : $L_1 \leftarrow L_1 + L_4$, $L_2 \leftarrow L_2 - L_4$, $L_5 \leftarrow -\frac{1}{3}(L_5 - L_4)$.

$$(S)\ \begin{cases} a-e &= 0 \\ b-f &= 0 \\ c+e+f &= 0 \\ d+e+f &= 0 \\ e+f &= 0 \end{cases}$$

Étape 5 : $L_1 \leftarrow L_1 + L_5$, $L_3 \leftarrow L_3 - L_5$, $L_4 \leftarrow L_4 - L_5$.

$$(S)\ \begin{cases} a+f &= 0 \\ b-f &= 0 \\ c &= 0 \\ d &= 0 \\ e+f &= 0 \end{cases}$$

Ainsi, $A \in \mathcal{S}$ et $\sigma(A) = 0$ si et seulement si

$$A = \begin{pmatrix} -f & f & 0 \\ f & 0 & -f \\ 0 & -f & f \end{pmatrix}$$

ou encore, en posant $\lambda = -f$, $A = \lambda V$, où $\lambda \in \mathbb{R}$ et

$$V = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & -1 \end{pmatrix}$$

Notons

$$U = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

On a $U \in \mathcal{S}$ et $\sigma(U) = 3$. Soit $A \in \mathcal{S}$. Soit $B = A - \frac{1}{3}\sigma(A)U$. On a alors $B \in \mathcal{S}$ et

$$\sigma(B) = \sigma(A) - \frac{1}{3}\sigma(A)\sigma(U) = 0$$

Ainsi, il existe $\lambda \in \mathbb{R}$ tel que $B = \lambda V$. De là,

$$A = \frac{1}{3}\sigma(A)U + \lambda V$$

Inversement, toute matrice de ce type est dans $\mathcal{S}$. Ainsi, $\mathcal{S} =< U, V >$ est de dimension 2.

1.2.4 2.3 Matrices magiques de taille 3

En regroupant le tout, on obtient $M_3 =< U, V, W >$. L'espace vectoriel M_3 est donc de dimension 3 : les matrices de E_3 sont les matrices de la forme

$$A = aU + bV + cW$$

où $a, b, c \in \mathbb{R}$. De plus, $a = \frac{1}{3}\sigma(A)$. En développant la combinaison linéaire ci-dessus, on obtient

$$A = \begin{pmatrix} a+b & a-b+c & a-c \\ a-b-c & a & a+b+c \\ a+c & a+b-c & a-b \end{pmatrix}$$

1.2.5 2.4 Les carrés magiques normaux de taille 3

Passons maintenant aux carrés magiques normaux. Pour un tel carré A, on a $\sigma(A) = 15$, et donc $a = 5$. Ainsi, en reprenant les notations ci-dessus,

$$A = \begin{pmatrix} 5+b & 5-b+c & 5-c \\ 5-b-c & 5 & 5+b+c \\ 5+c & 5+b-c & 5-b \end{pmatrix}$$

Il reste à déterminer b et c, qui sont clairement entiers. On a $1 \leq 5+b \leq 9$ donc $-4 \leq b \leq 4$. Il en est de même pour c. Nous pourrions faire le travail à la main, mais laissons faire Python.

[35]:
```
def carres_magiques_normaux_3():
    sols = []
    for b in range(-4, 5):
        for c in range(-4, 5):
            A = [[5 + b, 5 - b + c, 5 - c],[5 - b - c, 5, 5 + b + c],[5 + c, 5 +
 ↪b - c, 5 - b]]
            if est_carre_magique_normal(A):
                sols.append(A)
    return sols
```

[36]:
```
sols = carres_magiques_normaux_3()
print(len(sols))
```

8

Il y a ainsi 8 carrés magiques normaux de taille 3. Remarquons que, en appelant C l'un de ces carrés, tous les carrés sont de la forme fC où $f \in \mathbb{D}_4$.

Il existe donc essentiellement un seul carré magique normal de taille 3.

1.3 4. Carrés magiques normaux de taille impaire

Il existe de nombreuses méthodes pour créer des carrés magiques normaux de taille impaire. Nous allons décrire l'une d'entre-elles.

1.3.1 4.1 Une matrice magique

Soit $m \in \mathbb{N}^*$. Soit $n = 2m + 1$. Soit P la matrice de taille n dont les coefficients de la $i$ème ligne ($0 \leq i \leq n-1$) sont égaux à $i+1$: pour tous $i, j \in [|0, n-1|]$, $P_{ij} = i+1$. Soit U_n la matrice de taille n dont tous les coefficients sont égaux à 1. Posons

$$A = P + P^T + (m-1)U_n$$

et

$$B = P + 2P^T - 2U_n$$

Posons enfin

$$C = n(A \bmod n) + (B \bmod n) + U_n$$

où le modulo signifie que les coefficients des matrices A et B sont chacun pris modulo n. Quels sont les coefficients de la matrice C ? Soient $i, j \in [|0, n-1|]$. On a

$$A_{ij} = P_{ij} + P_{ji} + m - 1 = (i+1) + (j+1) + m - 1 = i + j + m + 1$$

$$B_{ij} = P_{ij} + 2P_{ji} - 2 = (i+1) + 2(j+1) - 2 = i + 2j + 1$$

et donc

$$C_{ij} = n((i+j+m+1) \bmod n) + ((i+2j+1) \bmod n) + 1$$

Proposition. La matrice C est magique.

Démonstration. Soit $i \in [|0, n-1|]$. Soient $j, j' \in [|0, n-1|]$. Supposons que $i + j + m + 1 \equiv i + j' + m + 1 \bmod n$. Alors, $j \equiv j' \bmod n$ et donc $j = j'$. De même, $i + 2j + 1 \equiv i + 2j' + 1 \bmod n$. Alors, $2j \equiv 2j' \bmod n$ et, comme n est impair, $j = j'$. Ainsi, les coefficients de la ligne i de $A \bmod n$ sont n entiers distincts entre 0 et $n-1$. Ce sont donc les entiers entre 0 et $n-1$. De là,

$$\sum_{j=0}^{n-1} A_{ij} \bmod n = \sum_{j=0}^{n-1} j = \frac{1}{2}n(n-1) = mn$$

De même,

$$\sum_{j=0}^{n-1} B_{ij} \bmod n = mn$$

et donc

$$\sum_{j=0}^{n-1} C_{ij} = mn^2 + mn + n = n(mn+m+1) = n(2m^2+2m+1) = \frac{1}{2}n(n^2+1)$$

Le même raisonnement montre que pour tout $j \in [|0, n-1|]$,

$$\sum_{i=0}^{n-1} C_{ij} = \frac{1}{2}n(n^2+1)$$

Passons aux diagonales de C. Pour tout $i \in [|0, n-1|]$,

$$A_{ii} = 2i + m + 1$$

Toujours par le même raisonnement,

$$\sum_{i=0}^{n-1} A_{ii} \bmod n = mn$$

On a aussi

$$B_{ii} = 3i + 1$$

Si n n'est pas multiple de 3, le raisonnement déjà fait plusieurs fois prouve que

$$\sum_{i=0}^{n-1} B_{ii} \bmod n = mn$$

Supposons maintenant que $n = 3p$ est multiple de 3. On a alors

$$\begin{array}{rcl}
\sum_{i=0}^{n-1} B_{ii} \bmod n & = & \sum_{i=0}^{3p-1}(3i+1) \bmod 3p \\
& = & 3\sum_{i=0}^{p-1}(3i+1) \\
& = & 3(\frac{3}{2}p(p-1)+p) \\
& = & \frac{3}{2}n(p-1)+n \\
& = & \frac{1}{2}n(n-1) \\
& = & nm
\end{array}$$

Dans tous les cas, on retrouve pour $\sum_{i=0}^{n-1} A_{ii} \bmod n$ et $\sum_{i=0}^{n-1} B_{ii} \bmod n$ les valeurs des sommes déjà trouvées pour les lignes de C. Ainsi,

$$\sum_{i=0}^{n-1} C_{ii} = \frac{1}{2}n(n^2+1)$$

Le calcul de la somme de la seconde diagonale de C est laissé en exercice.

1.3.2 4.2 C est un carré normal

Proposition. C est un carré magique normal.

Démonstration. Nous savons déjà que la matrice C est magique. Clairement, les coefficients de C sont des entiers naturels. Il reste à prouver que ce sont les entiers entre 1 et n^2.

Soient $i, j \in [|0, n-1|]$. Rappelons que

$$C_{ij} = n((i+j+m+1) \bmod n) + ((i+2j+1) \bmod n) + 1$$

Clairement, $C_{ij} \geq 1$. De plus,

$$C_{ij} \leq n(n-1) + (n-1) + 1 = n^2$$

Soient $i, j, i'j' \in [|0, n-1|]$. Posons $u = i+j+m+1$, $v = i+2j+1$, $u' = i'+j'+m+1$ et $v' = i'+2j'+1$. Supposons que $C_{ij} = C_{i'j'}$. On a donc

$$n(u \bmod n) + (v \bmod n) = n(u' \bmod n) + (v' \bmod n)$$

En prenant cette égalité modulo n, il vient $v \bmod n = v' \bmod n$. En reportant, on obtient

$$n(u \bmod n) = n(u' \bmod n)$$

d'où, en divisant par n, $u \bmod n = u' \bmod n$. On a donc

$$\left\{\begin{array}{lcl}
i+j+m+1 & \equiv & i'+j'+m+1 \bmod n \\
i+2j+1 & \equiv & i'+2j'+1 \bmod n
\end{array}\right.$$

d'où

$$\begin{cases} i+j & \equiv & i'+j' \bmod n \\ i+2j & \equiv & i'+2j' \bmod n \end{cases}$$

En soustrayant, il vient $j \equiv j' \bmod n$. Mais $j, j' \in [|0, n-1|]$, donc $j = j'$. En reportant, il vient $i \equiv i' \bmod n$ et donc, de même, $i = i'$.

En résumé, les coefficents de C sont n^2 entiers naturels distincts appartenant à $[|1, n^2|]$. Ce sont donc les entiers naturels entre 1 et n^2. Ainsi, C est un carré normal.

1.3.3 4.3 Le code Python

La fonction `carre_magique_normal_imair` prend en paramètre un entier impair $n \geq 3$ et renvoie un carré magique normal de taille n.

[37]:
```
def carre_magique_normal_impair(n):
    C = matrice_vide(n)
    m = (n - 1) // 2
    for i in range(n):
        for j in range(n):
            C[i][j] = n * ((i + j + m + 1) % n) + (i + 2 * j + 1) % n + 1
    return C
```

[38]:
```
C = carre_magique_normal_impair(11)
print(est_carre_magique_normal(C))
print_carre(C)
```

```
True
+----+----+----+----+----+----+----+----+----+----+----+
|  68|  81|  94| 107| 120|   1|  14|  27|  40|  53|  66|
+----+----+----+----+----+----+----+----+----+----+----+
|  80|  93| 106| 119|  11|  13|  26|  39|  52|  65|  67|
+----+----+----+----+----+----+----+----+----+----+----+
|  92| 105| 118|  10|  12|  25|  38|  51|  64|  77|  79|
+----+----+----+----+----+----+----+----+----+----+----+
| 104| 117|   9|  22|  24|  37|  50|  63|  76|  78|  91|
+----+----+----+----+----+----+----+----+----+----+----+
| 116|   8|  21|  23|  36|  49|  62|  75|  88|  90| 103|
+----+----+----+----+----+----+----+----+----+----+----+
|   7|  20|  33|  35|  48|  61|  74|  87|  89| 102| 115|
+----+----+----+----+----+----+----+----+----+----+----+
|  19|  32|  34|  47|  60|  73|  86|  99| 101| 114|   6|
+----+----+----+----+----+----+----+----+----+----+----+
|  31|  44|  46|  59|  72|  85|  98| 100| 113|   5|  18|
+----+----+----+----+----+----+----+----+----+----+----+
|  43|  45|  58|  71|  84|  97| 110| 112|   4|  17|  30|
+----+----+----+----+----+----+----+----+----+----+----+
```

```
|  55|  57|  70|  83|  96| 109| 111|   3|  16|  29|  42|
+----+----+----+----+----+----+----+----+----+----+----+
|  56|  69|  82|  95| 108| 121|   2|  15|  28|  41|  54|
+----+----+----+----+----+----+----+----+----+----+----+
```

1.4 5. Produit de carrés magiques normaux

1.4.1 5.1 Une opération sur les carrés magiques normaux

Soient A et B deux carrés magiques normaux de tailles respectives m et n. Nous allons définir une matrice $C = A \otimes B$ de taille mn. Le plus simple est de définir cette matrice par blocs. La matrice C est constituée de m^2 blocs de taille n. Pour $i, j \in [|0, m-1|]$, le bloc à la « ligne » i et à la « colonne » j de C est

$$C_{[ij]} = n^2(A_{ij} - 1)U_n + B$$

où U_n est la matrice de taille n qui ne contient que des 1.

Proposition. C est une matrice magique.

Démonstration. Soient $i, j \in [|0, m-1|]$. Soit $k \in [|0, n-1|]$. On a

$$\begin{array}{rcl}
\sum_{\ell=0}^{n-1}(C_{[ij]})_{k\ell} & = & \sum_{\ell=0}^{n-1}(n^2(A_{ij}-1)(U_n)_{k\ell} + B_{k\ell}) \\
& = & \sum_{\ell=0}^{n-1}(n^2(A_{ij}-1) + B_{k\ell}) \\
& = & n^3(A_{ij}-1) + \frac{1}{2}n(n^2+1) \\
& = & n^3 A_{ij} + \frac{1}{2}(n - n^3)
\end{array}$$

Soit $p \in [|0, mn-1|]$. p s'écrit de façon unique $p = ni + k$ où $i \in [|0, m-1|]$ et $k \in [|0, n-1|]$. La somme des coefficients de la ligne p de C est

$$\begin{array}{rcl}
\sum_{j=0}^{m-1}\sum_{\ell=0}^{n-1}(C_{[ij]})_{k\ell} & = & \sum_{j=0}^{m-1}(n^3 A_{ij} + \frac{1}{2}(n-n^3)) \\
& = & n^3\sigma(A) + \frac{1}{2}m(n-n^3)) \\
& = & \frac{1}{2}n^3 m(m^2+1) + \frac{1}{2}m(n-n^3)) \\
& = & \frac{1}{2}mn(m^2n^2+1)
\end{array}$$

Un calcul analogue montre que la somme des coefficients de C sur chacune de ses colonnes est aussi $\frac{1}{2}mn(m^2n^2+1)$. Il reste à vérifier la somme des coefficients sur chacune des diagonales. On a

$$\begin{array}{rcl}
\sum_{p=0}^{mn-1} C_{pp} & = & \sum_{j=0}^{m-1}\sum_{k=0}^{n-1}(C_{[jj]})_{kk} \\
& = & \sum_{j=0}^{m-1}\sum_{k=0}^{n-1}(n^2(A_{jj}-1) + B_{kk}) \\
& = & n\sum_{j=0}^{m-1} n^2(A_{jj}-1) + m\sum_{k=0}^{n-1} B_{kk} \\
& = & n(n^2\sigma(A) - mn^2) + m\sigma(B) \\
& = & n^3\frac{1}{2}m(m^2+1) - mn^3 + m\frac{1}{2}n(n^2+1) \\
& = & \frac{1}{2}mn(m^2n^2+1)
\end{array}$$

On a un résultat analogue pour l'autre diagonale. Ainsi, C est une matrice magique, de somme magique

$$\sigma(C) = \frac{1}{2}mn(m^2n^2 + 1)$$

Proposition. C est un carré magique normal.

Démonstration. Nous avons déjà vu que la matrice C est magique. Les coefficients de la matrice C sont clairement des entiers naturels, C est donc un carré magique. Il reste à montrer que les coefficients de C sont les entiers entre 1 et m^2n^2. Tout d'abord, soient $i, j \in [|0, m-1|]$ et $k, \ell \in [|0, n-1|]$. On a

$$(C_{[ij]})_{k\ell} = n^2(A_{ij} - 1) + B_{k\ell} \leq n^2(m^2 - 1) + n^2 = m^2n^2$$

et aussi

$$(C_{[ij]})_{k\ell} = n^2(A_{ij} - 1) + B_{k\ell} \geq B_{k\ell} \geq 1$$

Montrons maintenant que les coefficients de C sont distincts. Soient $i, j, i', j' \in [|0, m-1|]$ et $k, \ell, k', \ell' \in [|0, n-1|]$. Supposons que $(C_{[ij]})_{k\ell} = (C_{[i'j']})_{k'\ell'}$. On a donc

$$n^2(A_{ij} - 1) + B_{k\ell} = n^2(A_{i'j'} - 1) + B_{k'\ell'}$$

ou encore

$$n^2(A_{ij} - A_{i'j'}) = B_{k'\ell'} - B_{k\ell}$$

Le membre de droite de cette égalité est strictement compris entre $-n^2$ et n^2. Mais c'est aussi, en regardant le membre gauche, un multiple de n^2. Il en résulte qu'il est nul. Ainsi, $B_{k'\ell'} = B_{k\ell}$ et, en reportant, $A_{ij} = A_{i'j'}$. Comme les coefficients de A sont distincts deux à deux (et ceux de B aussi), il vient $k = k'$, $\ell = \ell'$, $i = i'$ et $j = j'$.

Résumons-nous : les coefficients de C sont m^2n^2 entiers distincts appartenant à $[|1, m^2n^2|]$. Ce sont donc **les** entiers de $[|1, m^2n^2|]$.

1.4.2 5.2 Le code

La fonction `produit` prend en paramètres deux carrés magiques normaux A et B. Elle renvoie $A \otimes B$.

[39]:
```
def produit(A, B):
    m, n = taille(A), taille(B)
    C = matrice_vide(m * n)
    for i in range(m * n):
        for j in range(m * n):
            C[i][j] = (A[i // n][j // n] - 1) * n ** 2 + B[i % n][j % n]
```

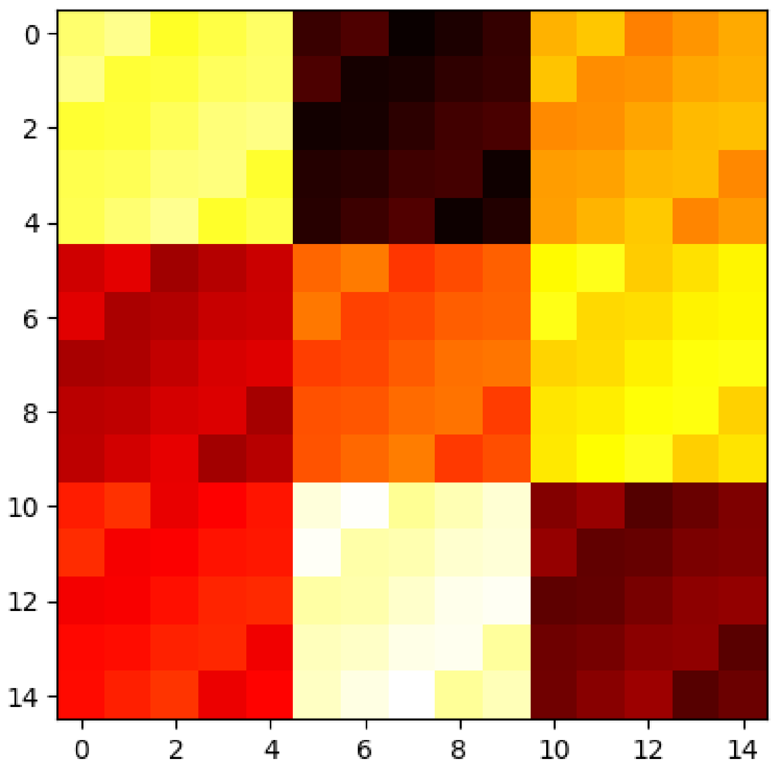
```python
    return C
```

Voici un carré magique normal de taille 15.

[40]:
```python
A = carre_magique_normal_impair(3)
B = carre_magique_normal_impair(5)
C = produit(A, B)
print(est_carre_magique_normal(C))
print_carre(C)
```

```
True
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
| 192| 199| 176| 183| 190|  17|  24|   1|   8|  15| 142| 149| 126| 133| 140|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
| 198| 180| 182| 189| 191|  23|   5|   7|  14|  16| 148| 130| 132| 139| 141|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
| 179| 181| 188| 195| 197|   4|   6|  13|  20|  22| 129| 131| 138| 145| 147|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
| 185| 187| 194| 196| 178|  10|  12|  19|  21|   3| 135| 137| 144| 146| 128|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
| 186| 193| 200| 177| 184|  11|  18|  25|   2|   9| 136| 143| 150| 127| 134|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  67|  74|  51|  58|  65| 117| 124| 101| 108| 115| 167| 174| 151| 158| 165|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  73|  55|  57|  64|  66| 123| 105| 107| 114| 116| 173| 155| 157| 164| 166|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  54|  56|  63|  70|  72| 104| 106| 113| 120| 122| 154| 156| 163| 170| 172|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  60|  62|  69|  71|  53| 110| 112| 119| 121| 103| 160| 162| 169| 171| 153|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  61|  68|  75|  52|  59| 111| 118| 125| 102| 109| 161| 168| 175| 152| 159|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  92|  99|  76|  83|  90| 217| 224| 201| 208| 215|  42|  49|  26|  33|  40|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  98|  80|  82|  89|  91| 223| 205| 207| 214| 216|  48|  30|  32|  39|  41|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  79|  81|  88|  95|  97| 204| 206| 213| 220| 222|  29|  31|  38|  45|  47|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  85|  87|  94|  96|  78| 210| 212| 219| 221| 203|  35|  37|  44|  46|  28|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  86|  93| 100|  77|  84| 211| 218| 225| 202| 209|  36|  43|  50|  27|  34|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
```

[41]:
```python
plot_matrice(C)
```



1.5 6. Carrés magiques normaux de taille paire

1.5.1 6.1 Introduction

Soit $m \geq 1$. Posons $u = 2m + 1$ et $n = 2u$. Nous allons fabriquer un carré magique normal de taille n à partir d'un carré magique normal de taille u. Nous serons donc en mesure de créer des carrés normaux de toutes les tailles qui sont le double d'un nombre impair. La méthode que nous allons décrire est due à Ralph Strachey (1868-1923).

Dans ce qui suit, A est un carré magique normal de taille u.

1.5.2 6.2 Première étape

Commençons par créer une matrice B de taille n. La matrice B est composée de 4 blocs de taille u. Pour $i, j \in \{0, 1\}$,

$$B_{[ij]} = A + \alpha_{ij} u^2 U_u$$

où U_u est la matrice de taille u qui ne contient que des 1, et

$$\alpha_{00} = 0, \alpha_{01} = 2, \alpha_{10} = 3, \alpha_{11} = 1$$

[42]:
```
def doubler(A):
    u = taille(A)
    n = 2 * u
    B = matrice_vide(n)
    for i in range(u):
        for j in range(u):
            B[i][j] = A[i][j]
            B[i][j + u] = A[i][j] + 2 * u ** 2
            B[i + u][j] = A[i][j] + 3 * u ** 2
            B[i + u][j + u] = A[i][j] + u ** 2
    return B
```

[43]:
```
A = carre_magique_normal_impair(3)
print_carre(A)
```

```
+----+----+----+
|   8|   1|   6|
+----+----+----+
|   3|   5|   7|
+----+----+----+
|   4|   9|   2|
+----+----+----+
```

[44]:
```
B = doubler(A)
print_carre(B)
```

```
+----+----+----+----+----+----+
|   8|   1|   6|  26|  19|  24|
+----+----+----+----+----+----+
|   3|   5|   7|  21|  23|  25|
+----+----+----+----+----+----+
|   4|   9|   2|  22|  27|  20|
+----+----+----+----+----+----+
|  35|  28|  33|  17|  10|  15|
+----+----+----+----+----+----+
|  30|  32|  34|  12|  14|  16|
+----+----+----+----+----+----+
|  31|  36|  29|  13|  18|  11|
+----+----+----+----+----+----+
```

Proposition. Les coefficients de B sont les entiers de $[|1, n^2|]$.

Démonstration. Soit $r \in [|1, n^2|]$.

- Si $1 \leq r \leq u^2$, alors r est un des coefficients de $A = B_{00}$.
- Si $u^2 + 1 \leq r \leq 2u^2$, alors $1 \leq r - u^2 \leq u^2$. Facilement, r est un des coefficients de B_{11}.
- Si $2u^2 + 1 \leq r \leq 3u^2$, alors $1 \leq r - 2u^2 \leq u^2$ et r est un des coefficients de B_{01}.
- Si $3u^2 + 1 \leq r \leq 4u^2$, alors $1 \leq r - 3u^2 \leq u^2$ et r est un des coefficients de B_{10}.

Tous les entiers de $[|1, n^2|]$ sont donc des coefficients de B. Comme la matrice B possède n^2 coefficients, ses coefficients sont **les** entiers de $[|1, n^2|]$.

1.5.3 6.3 Deuxième étape : échanges à droite

Dans les $m - 1$ dernières colonnes de B, on échange la première moitié et la deuxième moitié de la colonne. On crée ainsi une nouvelle matrice C. Plus formellement, soient $i, j \in [|0, n - 1|]$.

- Si $j \leq n - m$, $C_{ij} = B_{ij}$.
- Si $n - m + 1 \leq j < n$ et $i < u$, $C_{ij} = B_{(i+u)j}$.
- Si $n - m + 1 \leq j < n$ et $u \leq i < n$, $C_{ij} = B_{(i-u)j}$.

La fonction `echange_droit` ci-dessous fait le travail. Elle modifie B sur place.

[45]:
```
def echange_droit(B):
    n = taille(B)
```

```
    u = n // 2
    m = (u - 1) // 2
    for j in range(n - m + 1, n):
        for i in range(u):
            B[i][j] ,B[u + i][j] = B[u + i][j], B[i][j]
```

[46]:
```
A = carre_magique_normal_impair(5)
print_carre(A)
```

```
+----+----+----+----+----+
|  17|  24|   1|   8|  15|
+----+----+----+----+----+
|  23|   5|   7|  14|  16|
+----+----+----+----+----+
|   4|   6|  13|  20|  22|
+----+----+----+----+----+
|  10|  12|  19|  21|   3|
+----+----+----+----+----+
|  11|  18|  25|   2|   9|
+----+----+----+----+----+
```

[47]:
```
B = doubler(A)
print_carre(B)
```

```
+----+----+----+----+----+----+----+----+----+----+
|  17|  24|   1|   8|  15|  67|  74|  51|  58|  65|
+----+----+----+----+----+----+----+----+----+----+
|  23|   5|   7|  14|  16|  73|  55|  57|  64|  66|
+----+----+----+----+----+----+----+----+----+----+
|   4|   6|  13|  20|  22|  54|  56|  63|  70|  72|
+----+----+----+----+----+----+----+----+----+----+
|  10|  12|  19|  21|   3|  60|  62|  69|  71|  53|
+----+----+----+----+----+----+----+----+----+----+
|  11|  18|  25|   2|   9|  61|  68|  75|  52|  59|
+----+----+----+----+----+----+----+----+----+----+
|  92|  99|  76|  83|  90|  42|  49|  26|  33|  40|
+----+----+----+----+----+----+----+----+----+----+
|  98|  80|  82|  89|  91|  48|  30|  32|  39|  41|
+----+----+----+----+----+----+----+----+----+----+
|  79|  81|  88|  95|  97|  29|  31|  38|  45|  47|
+----+----+----+----+----+----+----+----+----+----+
|  85|  87|  94|  96|  78|  35|  37|  44|  46|  28|
+----+----+----+----+----+----+----+----+----+----+
|  86|  93| 100|  77|  84|  36|  43|  50|  27|  34|
+----+----+----+----+----+----+----+----+----+----+
```

[48]:
```
echange_droit(B)
print_carre(B)
```

```
+----+----+----+----+----+----+----+----+----+----+
|  17|  24|   1|   8|  15|  67|  74|  51|  58|  40|
+----+----+----+----+----+----+----+----+----+----+
|  23|   5|   7|  14|  16|  73|  55|  57|  64|  41|
+----+----+----+----+----+----+----+----+----+----+
|   4|   6|  13|  20|  22|  54|  56|  63|  70|  47|
+----+----+----+----+----+----+----+----+----+----+
|  10|  12|  19|  21|   3|  60|  62|  69|  71|  28|
+----+----+----+----+----+----+----+----+----+----+
|  11|  18|  25|   2|   9|  61|  68|  75|  52|  34|
+----+----+----+----+----+----+----+----+----+----+
|  92|  99|  76|  83|  90|  42|  49|  26|  33|  65|
+----+----+----+----+----+----+----+----+----+----+
|  98|  80|  82|  89|  91|  48|  30|  32|  39|  66|
+----+----+----+----+----+----+----+----+----+----+
|  79|  81|  88|  95|  97|  29|  31|  38|  45|  72|
+----+----+----+----+----+----+----+----+----+----+
|  85|  87|  94|  96|  78|  35|  37|  44|  46|  53|
+----+----+----+----+----+----+----+----+----+----+
|  86|  93| 100|  77|  84|  36|  43|  50|  27|  59|
+----+----+----+----+----+----+----+----+----+----+
```

1.5.4 6.3 Troisième étape : échanges à gauche

Dans les m premières colonnes de C, on échange la première moitié et la deuxième moitié de la colonne. Une exception à cela : dans la colonne 0 de C, on n'échange pas le coefficient de la ligne $m-1$ et le coefficient de la ligne $n-m$.

De plus, dans la colonne m de C, on échange le coefficient de la ligne m et le coefficient de la ligne $n-m-1$.

On crée ainsi une nouvelle matrice D. Plus formellement, soient $i, j \in [|0, n-1|]$.

- Si $m < j < n$, $D_{ij} = C_{ij}$.
- Si $1 \leq j < m$ et $i < u$, $D_{ij} = C_{(i+u)j}$.
- Si $1 \leq j < m$ et $u \leq i$, $D_{ij} = C_{(i-u)j}$.
- Si $i \neq m$, $D_{i0} = C_{(i+u)0}$.
- Si $i \neq n-m-1$, $D_{i0} = C_{(i-u)0}$.
- $D_{mm} = C_{(n-m-1)m}$.
- $D_{(n-m-1)m} = C_{mm}$.

La fonction `echange_gauche` ci-dessous fait le travail. Elle modifie C sur place.

[49]:
```
def echange_gauche(C):
    n = taille(C)
    u = n // 2
```

```
    m = (u - 1) // 2
    for j in range(m):
        for i in range(u):
            if (i, j) != (m, 0) and (i, j) != (u + m, 0):
                C[i][j] ,C[i + u][j] = C[i + u][j], C[i][j]
    C[m][m], C[u + m][m] = C[u + m][m], C[m][m]
```

[50]:
```
A = carre_magique_normal_impair(5)
print_carre(A)
```

```
+----+----+----+----+----+
|  17|  24|   1|   8|  15|
+----+----+----+----+----+
|  23|   5|   7|  14|  16|
+----+----+----+----+----+
|   4|   6|  13|  20|  22|
+----+----+----+----+----+
|  10|  12|  19|  21|   3|
+----+----+----+----+----+
|  11|  18|  25|   2|   9|
+----+----+----+----+----+
```

[51]:
```
B = doubler(A)
print_carre(B)
```

```
+----+----+----+----+----+----+----+----+----+----+
|  17|  24|   1|   8|  15|  67|  74|  51|  58|  65|
+----+----+----+----+----+----+----+----+----+----+
|  23|   5|   7|  14|  16|  73|  55|  57|  64|  66|
+----+----+----+----+----+----+----+----+----+----+
|   4|   6|  13|  20|  22|  54|  56|  63|  70|  72|
+----+----+----+----+----+----+----+----+----+----+
|  10|  12|  19|  21|   3|  60|  62|  69|  71|  53|
+----+----+----+----+----+----+----+----+----+----+
|  11|  18|  25|   2|   9|  61|  68|  75|  52|  59|
+----+----+----+----+----+----+----+----+----+----+
|  92|  99|  76|  83|  90|  42|  49|  26|  33|  40|
+----+----+----+----+----+----+----+----+----+----+
|  98|  80|  82|  89|  91|  48|  30|  32|  39|  41|
+----+----+----+----+----+----+----+----+----+----+
|  79|  81|  88|  95|  97|  29|  31|  38|  45|  47|
+----+----+----+----+----+----+----+----+----+----+
|  85|  87|  94|  96|  78|  35|  37|  44|  46|  28|
+----+----+----+----+----+----+----+----+----+----+
|  86|  93| 100|  77|  84|  36|  43|  50|  27|  34|
+----+----+----+----+----+----+----+----+----+----+
```

```
[52]: echange_gauche(B)
print_carre(B)
```

```
+----+----+----+----+----+----+----+----+----+----+
|  92|  99|   1|   8|  15|  67|  74|  51|  58|  65|
+----+----+----+----+----+----+----+----+----+----+
|  98|  80|   7|  14|  16|  73|  55|  57|  64|  66|
+----+----+----+----+----+----+----+----+----+----+
|   4|  81|  88|  20|  22|  54|  56|  63|  70|  72|
+----+----+----+----+----+----+----+----+----+----+
|  85|  87|  19|  21|   3|  60|  62|  69|  71|  53|
+----+----+----+----+----+----+----+----+----+----+
|  86|  93|  25|   2|   9|  61|  68|  75|  52|  59|
+----+----+----+----+----+----+----+----+----+----+
|  17|  24|  76|  83|  90|  42|  49|  26|  33|  40|
+----+----+----+----+----+----+----+----+----+----+
|  23|   5|  82|  89|  91|  48|  30|  32|  39|  41|
+----+----+----+----+----+----+----+----+----+----+
|  79|   6|  13|  95|  97|  29|  31|  38|  45|  47|
+----+----+----+----+----+----+----+----+----+----+
|  10|  12|  94|  96|  78|  35|  37|  44|  46|  28|
+----+----+----+----+----+----+----+----+----+----+
|  11|  18| 100|  77|  84|  36|  43|  50|  27|  34|
+----+----+----+----+----+----+----+----+----+----+
```

1.5.5 6.4 D est un carré magique normal

Proposition. La matrice D est un carré magique normal de taille n.

Démonstration. Nous avons déjà prouvé que B est un carré normal. De là, C, puis D le sont aussi. Il nous reste à montrer que la matrice D est magique.

Commençons par le plus simple, les somme des coefficients des colonnes de D. Comme les échanges faits pour passer de B à D ne modifient que l'ordre des coefficients dans les colonnes, il suffit de raisonner sur B. Soit $j \in [|0, n-1|]$. Supposons tout d'abord $j < u$.

On a

$$\begin{array}{rcl}
\sum_{i=0}^{n-1} D_{ij} & = & \sum_{i=0}^{n-1} B_{ij} \\
& = & \sum_{i=0}^{u-1} B_{ij} + \sum_{i=u}^{n-1} B_{ij} \\
& = & \sum_{i=0}^{u-1} A_{ij} + \sum_{i=0}^{u-1} (A_{ij} + 3u^2) \\
& = & 2\sigma(A) + 3u^3 \\
& = & u(u^2+1) + 3u^3 \\
& = & u(4u^2+1) \\
& = & \frac{1}{2} n(n^2+1)
\end{array}$$

Si $j \geq u$, un calcul analogue fournit le même résultat.

Passons aux lignes de D. Soit $i \in [|0, n-1|]$. Il y a un certain nombre de cas à considérer :

- $0 \leq i \leq u-1$ et $i \neq m$
- $i = m$
- $u \leq i \leq n-1$ et $i \neq n-1-m$
- $i = n-1-m$

Nous allons traiter les deux premiers cas.

Cas 1 : $0 \leq i \leq u-1$ et $i \neq m$. On a

$$\sum_{j=0}^{n-1} D_{ij} = \sum_{j=0}^{m-1} D_{ij} + \sum_{j=m}^{u-1} D_{ij} + \sum_{j=u}^{u+m+1} D_{ij} + \sum_{j=u+m+2}^{n-1} D_{ij}$$

La première somme est

$$\sum_{j=0}^{m-1} D_{ij} = \sum_{j=0}^{m-1} (A_{ij} + 3u^2) = \sum_{j=0}^{m-1} A_{ij} + 3mu^2$$

La deuxième somme est

$$\sum_{j=m}^{u-1} D_{ij} = \sum_{j=m}^{u-1} A_{ij}$$

La troisième somme est

$$\sum_{j=u}^{u+m+1} D_{ij} = \sum_{j=0}^{m+1} (A_{ij} + 2u^2) = \sum_{j=0}^{m+1} A_{ij} + 2(m+2)u^2$$

La quatrième somme est

$$\sum_{j=u+m+2}^{n-1} D_{ij} = \sum_{j=m+2}^{n-1} (A_{ij} + u^2) = \sum_{j=m+2}^{n-1} A_{ij} + (m-1)u^2$$

En ajoutant le tout, il vient

$$\begin{array}{rcl} \sum_{j=0}^{n-1} D_{ij} & = & \sum_{j=0}^{m-1} A_{ij} + 3mu^2 + \sum_{j=m}^{u-1} A_{ij} + \sum_{j=0}^{m+1} A_{ij} + 2(m+2)u^2 + \sum_{j=m+2}^{n-1} A_{ij} + (m-1)u^2 \\ & = & 2\sigma(A) + (6m+3)u^2 \\ & = & 2\sigma(A) + 3u^2 \\ & = & \frac{1}{2} n(n^2+1) \end{array}$$

Cas 2 : $i = m$. Les calculs sont à peu près identiques à ceux du cas 1, avec un petit décalage. On a, en reprenant certains des calculs faits dans le premier cas,

$$\begin{array}{rcl}
\sum_{j=0}^{n-1} D_{mj} & = & D_{m0} + \sum_{j=1}^{m} D_{mj} + \sum_{j=m+1}^{u-1} D_{mj} + \sum_{j=u}^{u+m+1} D_{mj} + \sum_{j=u+m+2}^{n-1} D_{mj} \\
& = & A_{m0} + (\sum_{j=1}^{m} A_{mj} + 3mu^2) + (\sum_{j=m+1}^{u-1} A_{ij}) + (\sum_{j=0}^{m+1} A_{ij} + 2(m+2)u^2) + (\sum_{j=m+2}^{n-1} A_{ij} + \\
& = & 2\sigma(A) + (6m+3)u^2 \\
& = & \frac{1}{2}n(n^2+1)
\end{array}$$

Pour finir, il reste à examiner les diagonales. Nous traitons la diagonale principale, le lecteur se fera une joie de faire les calculs pour l'autre diagonale. On a

$$\begin{array}{rcl}
\sum_{i=0}^{n-1} D_{ii} & = & \sum_{i=0}^{m} D_{ii} + \sum_{i=m+1}^{u-1} D_{ii} + \sum_{i=u}^{u+m+1} D_{ii} + \sum_{i=u+m+2}^{n-1} D_{ii} \\
& = & \sum_{i=0}^{m}(A_{ii} + 3u^2) + \sum_{i=m+1}^{u-1} A_{ii} + \sum_{i=0}^{m+1}(A_{ii} + u^2) + \sum_{i=m+2}^{u-1}(A_{ii} + 2u^2) \\
& = & 2\sigma(A) + (6m+3)u^2 \\
& = & \frac{1}{2}n(n^2+1)
\end{array}$$

1.5.6 6.5 Le code final

La fonction `carre_magique_normal_pair` prend en paramètre un entier n de la forme $2(2m+1)$ où $m \geq 1$. Elle renvoie un carré magique normal de taille n.

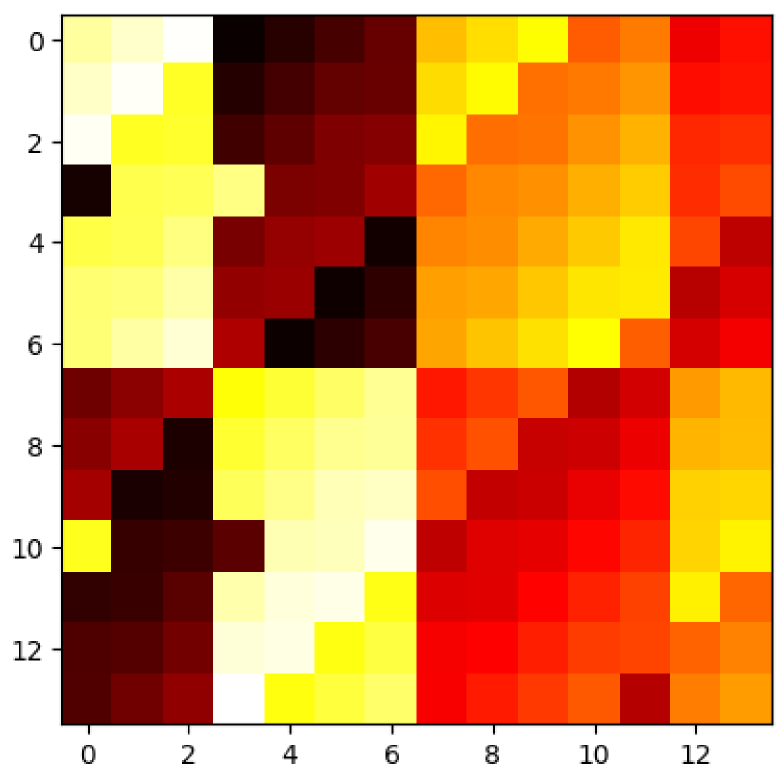
[53]:
```
def carre_magique_normal_pair(n):
    u = n // 2
    A = carre_magique_normal_impair(u)
    B = doubler(A)
    echange_droit(B)
    echange_gauche(B)
    return B
```

[54]:
```
C = carre_magique_normal_pair(14)
print(est_carre_magique_normal(C))
print_carre(C)
```

```
True
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
| 177| 186| 195|   1|  10|  19|  28| 128| 137| 146|  99| 108|  68|  77|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
| 185| 194| 154|   9|  18|  27|  29| 136| 145| 105| 107| 116|  76|  78|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
| 193| 153| 155|  17|  26|  35|  37| 144| 104| 106| 115| 124|  84|  86|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|   5| 161| 163| 172|  34|  36|  45| 103| 112| 114| 123| 132|  85|  94|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
| 160| 162| 171|  33|  42|  44|   4| 111| 113| 122| 131| 140|  93|  53|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
| 168| 170| 179|  41|  43|   3|  12| 119| 121| 130| 139| 141|  52|  61|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
```

```
| 169| 178| 187|  49|   2|  11|  20| 120| 129| 138| 147| 100|  60|  69|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  30|  39|  48| 148| 157| 166| 175|  79|  88|  97|  50|  59| 117| 126|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  38|  47|   7| 156| 165| 174| 176|  87|  96|  56|  58|  67| 125| 127|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  46|   6|   8| 164| 173| 182| 184|  95|  55|  57|  66|  75| 133| 135|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
| 152|  14|  16|  25| 181| 183| 192|  54|  63|  65|  74|  83| 134| 143|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  13|  15|  24| 180| 189| 191| 151|  62|  64|  73|  82|  91| 142| 102|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  21|  23|  32| 188| 190| 150| 159|  70|  72|  81|  90|  92| 101| 110|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|  22|  31|  40| 196| 149| 158| 167|  71|  80|  89|  98|  51| 109| 118|
+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
```

[55]:
```
plot_matrice(C)
```



1.6 7. Le cas général

Théorème. Soit $n \in \mathbb{N}^*$ un entier différent de 2. Il existe un carré magique normal de taille n.

Il est en effet maintenant facile de créer des carrés magiques normaux de toutes les tailles (sauf 2, évidemment). Soit $n \geq 1$.

- Si $n = 1$, c'est facile.
- Si $n \geq 3$ est impair, on sait fabriquer un carré normal de taille n.
- Si $n \geq 4$ est le double d'un nombre impair, on sait également faire.
- Si $n \geq 4$ est un multiple de 4, écrivons $n = 4^k m$, où $k \geq 1$ et $m \geq 1$ n'est pas multiple de 4. Soit A un carré magique normal de taille 4.
  - Si $m \neq 2$, alors m est impair ou m est le double d'un entier impair différent de 1. Soit B un carré magique normal de taille m. Alors, $C = A \otimes ... \otimes A \otimes B$ (où A apparaît k fois) est un carré magique normal de taille n.
  - Si $m = 2$, alors $n = 4^k \times 2 = 4^{k-1} \times 8$. Soit B un carré magique normal de taille 8. Alors, $C = A \otimes ... \otimes A \otimes B$ (où A apparaît $k - 1$ fois) est un carré magique normal de taille n.

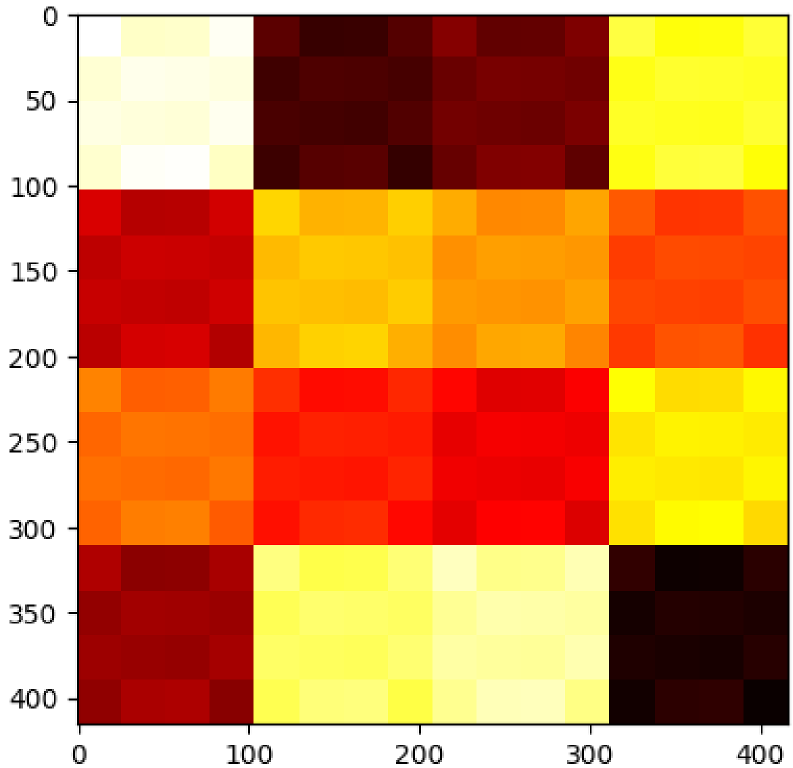
[56]:
```
def carre_magique_normal(n):
    if n == 1: return [[1]]
    elif n == 2: raise Exception('Pas de carré normal d\'ordre 2')
    elif n % 2 == 1: return carre_magique_normal_impair(n)
    elif n % 4 == 2: return carre_magique_normal_pair(n)
    else:
        k = 0
        while n % 4 == 0:
            k += 1
            n = n // 4
        C = [[1]]
        for i in range(k - 1): C = produit(C, carre4)
        if n == 2:
            return produit(C, carre8)
        else:
            C = produit(C, carre4)
            if n % 2 == 0: return produit(C, carre_magique_normal_pair(n))
            else: return produit(C, carre_magique_normal_impair(n))
```

Dans la cellule ci-dessous, on appelle `carre_magique_normal` pour tous les entiers n entre 3 et 150. On vérifie que les résultats sont effectivement des carrés magiques normaux. L'évaluation de la cellule devrait simplement afficher `ok`.

[57]:
```
for n in range(3, 150):
    C = carre_magique_normal(n)
    if not est_carre_magique_normal(C):
        print(n, end = ' ')
print('ok')
```

```
ok
```

[58]:
```
plot_matrice(carre_magique_normal(32 * 13))
```



1.7 Appendice - Combien de carrés magiques ?

Il y aurait encore beaucoup de choses à dire sur le sujet, mais ce notebook est déjà d'une longueur terrible. Signalons, pour terminer, que pour $n \geq 6$, le nombre de carrés magiques normaux de taille n n'est pas connu à l'heure actuelle. En notant C_n ce nombre, on a les valeurs suivantes pour $n \leq 5$.

n	1	2	3	4	5
C_n	1	0	8	7040	2202441792

On a obtenu, par des méthodes probabilistes, une estimation de C_n pour des valeurs de n supérieures. Pour n'en citer que deux,

$$C_6 \simeq 1.42 \times 10^{20}$$

$$C_{30} \simeq 5.3 \times 10^{2057}$$

1.8 Et maintenant ?

Nous avons dans ce notebook décrit un algorithme permettant, pour tout entier $n \geq 3$, de déterminer un carré magique normal de taille n. Cela dit, pour un entier n fixé, notre algorithme renvoie toujours le même carré, ce qui est un peu frustrant : notre algorithme est **déterministe**. Serait-il possible d'écrire un algorithme **randomisé** qui renvoie à chaque appel des carrés magiques différents ? La réponse est oui. Dans le notebook suivant nous verrons comment l'algorithme du **recuit simulé** permet de faire cela.