

```

1 import numpy as np
2
3
4 def encode_matrix(array):
5     new_matrix = []
6     # initialize array size
7     row = len(array)
8     col = len(array[0])
9
10    # matrix encode
11    for i in range(0, row):
12        new_array = []
13        for j in range(0, col):
14
15            # first and last row and column ignored
16            if i in (0, row - 1) or j in (0, col - 1):
17                new_array.insert(len(new_array), matrix[i][j])
18            else:
19                # 3x3 surrounding matrix
20                matrixOfThree = [[matrix[i - 1][j - 1] - matrix[i][j],
21                                matrix[i - 1][j] - matrix[i][j],
22                                matrix[i - 1][j + 1] - matrix[i][j]],
23                                [matrix[i][j - 1] - matrix[i][j],
24                                matrix[i][j], matrix[i][j + 1] - matrix
25                                [i][j]],
26                                [matrix[i + 1][j - 1] - matrix[i][j],
27                                matrix[i + 1][j] - matrix[i][j],
28                                matrix[i + 1][j + 1] - matrix[i][j]]]
29
30                # setting binary numbers
31                for k in range(0, 3):
32                    for l in range(0, 3):
33                        if k != 1 or l != 1:
34                            if matrixOfThree[k][l] >= 0:
35                                matrixOfThree[k][l] = 1
36                            else:
37                                matrixOfThree[k][l] = 0
38
39                # binary string
40                binary_string = str(matrixOfThree[0][0]) \
41                                + str(matrixOfThree[1][0]) \
42                                + str(matrixOfThree[2][0]) \
43                                + str(matrixOfThree[2][1]) \
44                                + str(matrixOfThree[2][2]) \
45                                + str(matrixOfThree[1][2]) \
46                                + str(matrixOfThree[0][2]) \
47                                + str(matrixOfThree[0][1])
48
49                # getting the integer value
50                integer_val = int(binary_string, 2)
51                # input the value to the new matrix

```

```
50         new_array.insert(len(new_array), integer_val)
51         new_matrix.insert(len(new_matrix), new_array)
52     return new_matrix
53
54
55 # 9x9 matrix
56 matrix = np.array(
57     [[0, 3, 2, 5, 4, 7, 6, 9, 8],
58
59      [3, 0, 1, 2, 3, 4, 5, 6, 7],
60
61      [2, 1, 0, 3, 2, 5, 4, 7, 6],
62
63      [5, 2, 3, 0, 1, 2, 3, 4, 5],
64
65      [4, 3, 2, 1, 0, 3, 2, 5, 4],
66
67      [7, 4, 5, 2, 3, 0, 1, 2, 3],
68
69      [6, 5, 4, 3, 2, 1, 0, 3, 2],
70
71      [9, 6, 7, 4, 5, 2, 3, 0, 1],
72
73      [8, 7, 6, 5, 4, 3, 2, 1, 0]])
74
75 print(matrix)
76 print("\n")
77 # encoding the matrix
78
79 new_matrix = np.array(encode_matrix(matrix))
80 # encoded matrix
81 print('encoded matrix: ')
82 for k in range(0, len(new_matrix)):
83     for l in range(0, len(new_matrix[0])):
84         print(str(new_matrix[k][l]), end=' ')
85     print(' ')
86
```