**Code:**

import numpy as np

from pprint import pprint

from collections import defaultdict

m = np.matrix(

[

[1, 2, 0],

[3, 0, 5],

[0, 7, 0],

[4, 0, 0],

[1, 8, 2]

]

)

n = np.matrix(

[

[1, 4, 5, 0],

[0, 4, 0, 1],

[6, 3, 9, 1],

]

)

shape\_m = m.shape

shape\_n = n.shape

m\_n = m \* n

pprint(m\_n)

m\_n = [[i, j, m\_n[i,j]] for i in range(m\_n.shape[0]) for j in range(m\_n.shape[1]) if m\_n[i,j] != 0]

pprint(m\_n)

m = [[i, j, m[i,j]] for i in range(m.shape[0]) for j in range(m.shape[1]) if m[i,j] != 0]

n = [[i, j, n[i,j]] for i in range(n.shape[0]) for j in range(n.shape[1]) if n[i, j] != 0]

pprint(m)

pprint(n)

ma = defaultdict(list)

for j in range(len(m)):

ma[m[j][1]].append((m[j][0], m[j][2]))

# key = j, value = njk

na = defaultdict(list)

for j in range(len(n)):

na[n[j][0]].append((n[j][1], n[j][2]))

pprint(ma)

pprint(na)

op = defaultdict(list)

for j in range(shape\_m[1]):

if j in ma and j in na:

for mi in ma[j]:

for ni in na[j]:

i = mi[0]

k = ni[0]

op[(i,k)].append(mi[1] \* ni[1])

pprint(op)

ans = list()

for k, v in op.items():

ans.append([k[0], k[1], sum(v)])

pprint(sorted(ans))

pprint(m\_n)

**OUTPUT :**

matrix([[ 1, 12, 5, 2],

[33, 27, 60, 5],

[ 0, 28, 0, 7],

[ 4, 16, 20, 0],

[13, 42, 23, 10]])

[[0, 0, 1],

[0, 1, 12],

[0, 2, 5],

[0, 3, 2],

[1, 0, 33],

[1, 1, 27],

[1, 2, 60],

[1, 3, 5],

[2, 1, 28],

[2, 3, 7],

[3, 0, 4],

[3, 1, 16],

[3, 2, 20],

[4, 0, 13],

[4, 1, 42],

[4, 2, 23],

[4, 3, 10]]

defaultdict(<class 'list'>,

{0: [(0, 1), (1, 3), (3, 4), (4, 1)],

1: [(0, 2), (2, 7), (4, 8)],

2: [(1, 5), (4, 2)]})

defaultdict(<class 'list'>,

{0: [(0, 1), (1, 4), (2, 5)],

1: [(1, 4), (3, 1)],

2: [(0, 6), (1, 3), (2, 9), (3, 1)]})

defaultdict(<class 'list'>,

{(0, 0): [1],

(0, 1): [4, 8],

(0, 2): [5],

(0, 3): [2],

(1, 0): [3, 30],

(1, 1): [12, 15],

(1, 2): [15, 45],

(1, 3): [5],

(2, 1): [28],

(2, 3): [7],

(3, 0): [4],

(3, 1): [16],

(3, 2): [20],

(4, 0): [1, 12],

(4, 1): [4, 32, 6],

(4, 2): [5, 18],

(4, 3): [8, 2]})