```
#strassen's matrix mul
import numpy as np

def split(mat):
    r,c=mat.shape
    r2,c2=r//2,c//2
    return
mat[:r2,:c2],mat[:r2,c2:],mat[r2:,:c2],mat[r2:,c2:]

#base case when size of mat is 1X1
def strassen(A,B):
    if len(A)==1:
        return A*B

    #splitting the matrices into quadrants(each of 4
matrices),this will be donerecursively untill the base case
is reached
    a,b,c,d=split(A)   #a11=a,a12=b,a21=c,a22=d
    e,f,g,h=split(B)   #b11=e,b12=f,b21=g,b22=h

    #computing the 7 products,recursively(p1,p2....p7)
    p1=strassen(a,f-h)
    p2=strassen(a+b,h)
    p3=strassen(c+d,e)
    p4=strassen(d,g-e)
    p5=strassen(a+d,e+h)
    p6=strassen(b-d,g+h)
    p7=strassen(a-c,e+f)

    #computing the values of 4 quadrants of the final matrix
c
    c11=p5+p4-p2+p6
    c12=p1+p2
    c21=p3+p4
    c22=p1+p5-p3-p7

    #combining the 4 quadrants into a single mat by stacking
horizontally & vertically
```

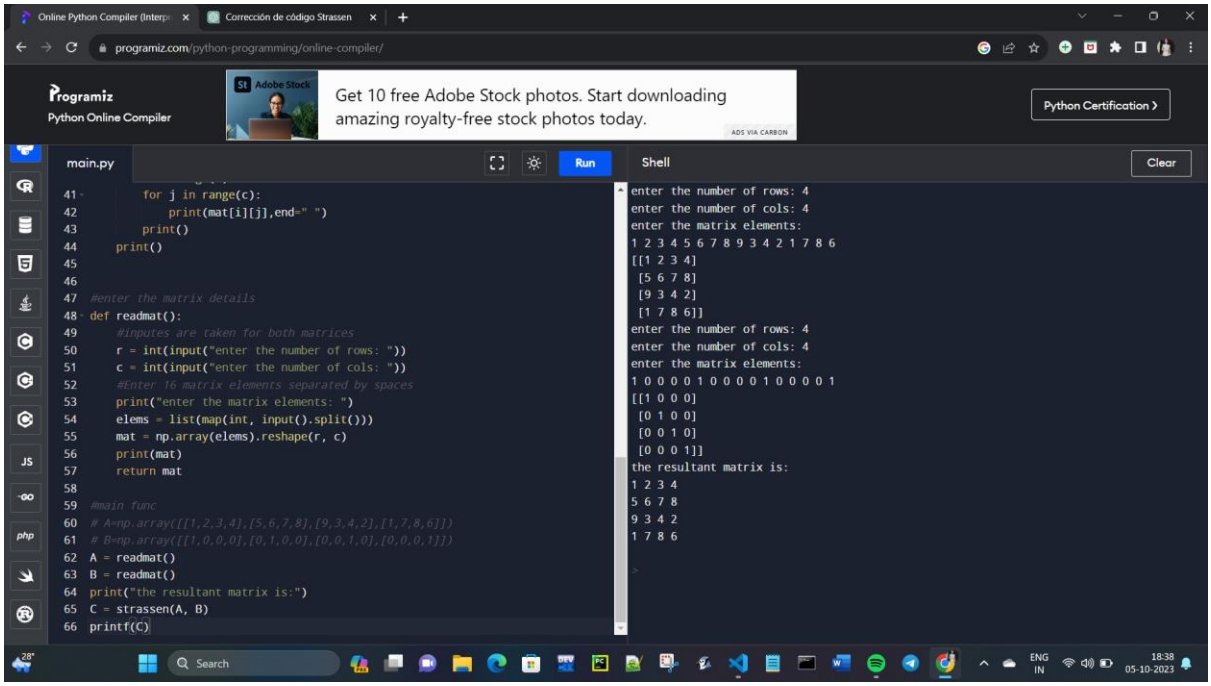
```
    c=np.vstack((np.hstack((c11,c12)),np.hstack((c21,c22))))

    return c

def printf(mat):
    r,c=mat.shape
    for i in range(c):
        for j in range(c):
            print(mat[i][j],end=" ")
        print()
    print()


#enter the matrix details
def readmat():
    #inputes are taken for both matrices
    r = int(input("enter the number of rows: "))
    c = int(input("enter the number of cols: "))
    #Enter 16 matrix elements separated by spaces
    print("enter the matrix elements: ")
    elems = list(map(int, input().split()))
    mat = np.array(elems).reshape(r, c)
    print(mat)
    return mat

#main func
#
A=np.array([[1,2,3,4],[5,6,7,8],[9,3,4,2],[1,7,8,6]])
# B=np.array([[1,0,0,0],[0,1,0,0],[0,0,1,0],[0,0,0,1]])
A = readmat()
B = readmat()
print("the resultant matrix is:")
C = strassen(A, B)
printf(C)
```


Online Python Compiler (Interp
Corrección de código Strassen
programiz.com/python-programming/online-compiler/
Programiz
Python Online Compiler
Get 10 free Adobe Stock photos. Start downloading amazing royalty-free stock photos today.
ADS VIA CARBON
Python Certification
main.py
Run
Shell
Clear
for j in range(c):
print(mat[i][j],end=" ")
print()
print()
#enter the matrix details
def readmat():
#inputes are taken for both matrices
r = int(input("enter the number of rows: "))
c = int(input("enter the number of cols: "))
#Enter 16 matrix elements separated by spaces
print("enter the matrix elements: ")
elems = list(map(int, input().split()))
mat = np.array(elems).reshape(r, c)
print(mat)
return mat
#main func
# A=np.array([[1,2,3,4],[5,6,7,8],[9,3,4,2],[1,7,8,6]])
# B=np.array([[1,0,0,0],[0,1,0,0],[0,0,1,0],[0,0,0,1]])
A = readmat()
B = readmat()
print("the resultant matrix is:")
C = strassen(A, B)
printf(C)
enter the number of rows: 4
enter the number of cols: 4
enter the matrix elements:
1 2 3 4 5 6 7 8 9 3 4 2 1 7 8 6
[[1 2 3 4]
[5 6 7 8]
[9 3 4 2]
[1 7 8 6]]
enter the number of rows: 4
enter the number of cols: 4
enter the matrix elements:
1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1
[[1 0 0 0]
[0 1 0 0]
[0 0 1 0]
[0 0 0 1]]
the resultant matrix is:
1 2 3 4
5 6 7 8
9 3 4 2
1 7 8 6
Search
ENG
IN
18:38
05-10-2023