

Matrix Multiplication using Strassen's Method

Code:

```
import math

arr1=[]

arr2=[]

arr3=[]

xlist=[]

def onedlist(a):

    if isinstance(a[0], list)==True:

        for i in a:

            onedlist(i)

    else:

        xlist.append(a)

def addm(a,b,l):

    leng = len(a)

    x=[]

    if l>2:

        for k in range(0, leng):

            for i in range(0, len(a[k])):

                for j in range(0, len(a[k][i])):

                    for l in range(0, len(a[k][i][j])):

                        a[k][i][j][l]=a[k][i][j][l]+b[k][i][j][l]

    return a;
```

else:

for i in range(0, leng):

t=[]

for j in range(0, leng):

t.append(a[i][j]+b[i][j])

x.append(t)

return x

def dividematrix(a):

leng = len(a)

mid = int(leng/2)

a11=[]

a12=[]

a21=[]

a22=[]

for i in range(0,mid):

t=[]

for j in range(0,mid):

t.append(a[i][j])

a11.append(t)

tt=[]

for j in range(mid, leng):

tt.append(a[i][j])

a12.append(tt)

for i in range(mid, leng):

Devang Chhajed

```
t=[]

for j in range(0,mid):
    t.append(a[i][j])
a21.append(t)

tt=[]

for j in range(mid, leng):
    tt.append(a[i][j])
a22.append(tt)

if len(a11[0])<=1:
    n=[[a11[0][0],a12[0][0]],[a21[0][0],a22[0][0]]]
else:
    n=[[a11,a12],[a21,a22]]

#print("div fun",n)

return n

#def mulmat(a,b,ax1,ax2,ay1,ay2,bx1,bx2,by1,by2):
def mulmat(a,b, l):

    alla = dividematrix(a)
    allb = dividematrix(b)

    if l > 2:

        #a
        a11=alla[0][0]
        a12=alla[0][1]
```

Devang Chhajed

```
a21=alla[1][0]
```

```
a22=alla[1][1]
```

```
#b
```

```
b11=allb[0][0]
```

```
b12=allb[0][1]
```

```
b21=allb[1][0]
```

```
b22=allb[1][1]
```

```
ae=mulmat(a11,b11,l/2)
```

```
bg=mulmat(a12,b21,l/2)
```

```
af=mulmat(a11,b12,l/2)
```

```
bh=mulmat(a12,b22,l/2)
```

```
ce=mulmat(a21,b11,l/2)
```

```
dg=mulmat(a22,b21,l/2)
```

```
cf=mulmat(a21,b12,l/2)
```

```
dh=mulmat(a22,b22,l/2)
```

```
c11 =addm(mulmat(a11,b11,l/2), mulmat(a12,b21,l/2),l/2)
```

```
c12 =addm(mulmat(a11,b12,l/2), mulmat(a12,b22,l/2),l/2)
```

```
c21 =addm(mulmat(a21,b11,l/2), mulmat(a22,b21,l/2),l/2)
```

```
c22 =addm(mulmat(a21,b12,l/2), mulmat(a22,b22,l/2),l/2)
```

```
t=[[c11,c12],[c21,c22]]
```

```
return t
```

else:

Devang Chhajed

```
"""
```

```
print(a,"-",b)
```

```
print("Star",strassens(a,b))
```

```
print("-----")
```

```
"""
```

```
return strassens(a,b)
```

```
def strassens(a, b):
```

```
    P=(a[0][0]+a[1][1])*(b[0][0]+b[1][1])
```

```
    Q=(a[1][0]+a[1][1])*b[0][0]
```

```
    R=a[0][0]*(b[0][1]-b[1][1])
```

```
    S=a[1][1]*(b[1][0]-b[0][0])
```

```
    T=(a[0][0]+a[0][1])*b[1][1]
```

```
    U=(a[1][0]-a[0][0])*(b[0][0]+b[0][1])
```

```
    V=(a[0][1]-a[1][1])*(b[1][0]+b[1][1])
```

```
    x=[[0,0],[0,0]]
```

```
    x[0][0]=P+S-T+V
```

```
    x[0][1]=R+T
```

```
    x[1][0]=Q+S
```

```
    x[1][1]=P+R-Q+U
```

```
    return x
```

```
s = int(input("Enter Matrix size : "))
```

Devang Chhajed

```
tarr1=[]
```

```
print("Enter Matrix 1")
```

```
for i in range(0,s):
```

```
    a=list(map(int, input().split()))
```

```
    tarr1.append(a)
```

```
tarr2=[]
```

```
print("Enter Matrix 2")
```

```
for i in range(0,s):
```

```
    a=list(map(int, input().split()))
```

```
    tarr2.append(a)
```

```
size=s
```

```
if math.log(s,2)-int(math.log(s,2))!=0.0 and s>2:
```

```
    size = 2**((int(math.log(s,2))+1)
```

```
arr1=[]
```

```
arr2=[]
```

```
arr3=[]
```

```
for i in range(0,size):
```

```
    l1=[]
```

```
    l2=[]
```

```
    l3=[]
```

```
    for j in range(0,size):
```

```
        l1.append(0)
```

```
        l2.append(0)
```

```
        l3.append(0)
```

Devang Chhajed

```
arr1.append(l1)
```

```
arr2.append(l2)
```

```
arr3.append(l3)
```

```
for i in range(0,len(tarr1)):
```

```
    for j in range(0,len(tarr2)):
```

```
        arr1[i][j]=tarr1[i][j]
```

```
        arr2[i][j]=tarr2[i][j]
```

```
x = mulmat(arr1,arr2,size)
```

```
print("Divide and Conquer -- ")
```

```
for r in x:
```

```
    print(r)
```

```
tarr3=[]
```

```
for i in range(0,s):
```

```
    ll=[]
```

```
    for j in range(0,s):
```

```
        ll.append(0)
```

```
    tarr3.append(ll)
```

```
# iterate through rows of X
```

```
for i in range(s):
```

```
    # iterate through columns of Y
```

```
    for j in range(s):
```

Devang Chhajed

```
# iterate through rows of Y
for k in range(s):
    tarr3[i][j] += tarr1[i][k] * tarr2[k][j]

print("Verify -- ")
for r in tarr3:
    print(r)

#onedlist(x)
#print(len(xlist))

#mul4(arr1, arr2)
#mulmat(arr1,arr2,0, len(arr1),0, len(arr1),0,len(arr2),0,len(arr2))
#arr3 = strassens(arr1, arr2)

#printarr(arr3)
```

Output:

Python 3.6.7 (v3.6.7:6ec5cf24b7, Oct 20 2018, 13:35:33) [MSC v.1900 64 bit (AMD64)] on win32

Type "help", "copyright", "credits" or "license()" for more information.

>>>

===== RESTART: C:\Users\Devang Chhajed\Desktop\matrixdnc.py =====

Enter Matrix size : 2

Enter Matrix 1

Devang Chhajed

7 6

8 5

Enter Matrix 2

7 11

8 12

Divide and Conquer --

[97, 149]

[96, 148]

Verify --

[97, 149]

[96, 148]

>>>

===== RESTART: C:\Users\Devang Chhajed\Desktop\matrixdnc.py =====

Enter Matrix size : 3

Enter Matrix 1

1 2 3

4 5 6

7 8 9

Enter Matrix 2

9 8 7

6 5 4

3 2 1

Divide and Conquer --

[[[30, 24], [84, 69]], [[18, 0], [54, 0]]]

[[[138, 114], [0, 0]], [[90, 0], [0, 0]]]

Verify --

[30, 24, 18]

[84, 69, 54]

Devang Chhajed

[138, 114, 90]

>>>

===== RESTART: C:\Users\Devang Chhajed\Desktop\matrixdnc.py =====

Enter Matrix size : 4

Enter Matrix 1

7 6 9 12

8 5 3 2

6 5 12 15

5 3 10 4

Enter Matrix 2

7 11 15 19

8 12 16 20

9 13 17 21

10 14 18 22

Divide and Conquer --

[[[298, 434], [143, 215]], [[570, 706], [287, 359]]]

[[[340, 492], [189, 277]], [[644, 796], [365, 453]]]

Verify --

[298, 434, 570, 706]

[143, 215, 287, 359]

[340, 492, 644, 796]

[189, 277, 365, 453]

>>>

===== RESTART: C:\Users\Devang Chhajed\Desktop\matrixdnc.py =====

Enter Matrix size : 8

Enter Matrix 1

1 2 3 4 5 6 7 8

9 8 7 6 5 4 3 2

Devang Chhajed

2 5 8 9 6 3 2 1

4 5 6 9 8 7 5 2

3 2 1 4 5 6 9 8

4 1 2 5 8 9 6 3

3 2 1 4 7 8 5 2

3 2 1 4 5 6 9 8

Enter Matrix 2

9 6 3 2 1 4 7 5

6 5 4 1 2 3 9 8

3 2 5 8 7 4 1 2

3 6 9 8 5 3 7 8

6 8 2 1 3 7 5 2

7 2 3 6 1 8 9 1

2 3 7 8 5 2 1 4

3 6 8 4 2 5 6 3

Divide and Conquer --

[[[[[152, 167], [238, 213]], [[203, 189], [207, 191]]], [[163, 173], [224, 220]], [[190, 189], [231, 231]]],
[[[118, 171], [142, 189]], [[190, 127], [260, 203]]], [[146, 157], [161, 214]], [[195, 164], [262, 193]]]]

[[[[[168, 181], [195, 181]], [[213, 193], [180, 187]]], [[168, 153], [168, 181]], [[147, 151], [213, 193]]],
[[[116, 175], [114, 197]], [[204, 141], [219, 130]]], [[92, 167], [116, 175]], [[192, 113], [204, 141]]]]

Verify --

[152, 167, 203, 189, 118, 171, 190, 127]

[238, 213, 207, 191, 142, 189, 260, 203]

[163, 173, 190, 189, 146, 157, 195, 164]

[224, 220, 231, 231, 161, 214, 262, 193]

[168, 181, 213, 193, 116, 175, 204, 141]

[195, 181, 180, 187, 114, 197, 219, 130]

[168, 153, 147, 151, 92, 167, 192, 113]

[168, 181, 213, 193, 116, 175, 204, 141]

Devang Chhajed

>>>