Name: Krish Sukhani

UID: 2018140059

Batch: D

Branch: IT

```
import numpy as np
import math


def alpha_calc(N):
  alpha = np.zeros(N, dtype=float)
  for u in range(N):
    if u == 0:
      alpha[u] = math.sqrt(1/N)
    else:
      alpha[u] = math.sqrt(2/N)
  return alpha


def dct_matrix_calc(N):
  C = np.zeros([N,N], dtype=float)
  alpha = alpha_calc(N)
  for u in range(N):
    for v in range(N):
      C[u][v] = round(alpha[u]*math.cos(((math.pi/2)*((2*v+1)*u)/N)),3)
  return C
```

Formula to find DCT of image : DCT = CFC'

```
def calculate_DCT_of_img(F):
  C = dct_matrix_calc(F.shape[0])
  print('The DCT Matrix :')
  print(C)
  D = np.dot(C,F)
  final_DCT = np.dot(D,C.T)
  print('The final DCT of the given image is :')
  print(final_DCT)
```

```
F = np.array([[2,4,4,2],[4,6,8,3],[2,8,10,4],[3,8,6,2]])
```

```
calculate_DCT_of_img(F)
```

```
The DCT Matrix :
[[ 0.5    0.5    0.5    0.5  ]
 [ 0.653  0.271 -0.271 -0.653]
 [ 0.5   -0.5   -0.5    0.5  ]
 [ 0.271 -0.653  0.653 -0.271]]
The final DCT of the given image is :
[[19.       -0.271    -8.        0.653   ]
 [-2.692    -0.249446  2.31      0.896178]
 [-3.5       1.466     1.5      -1.688   ]
 [ 0.031    -1.603072 -0.955    -0.250404]]
```

Conclusion:

In this experiment we successfully computed the DCT Matrix for N samples using the DCT equation DCT of the given input image.