

**ΜΥΕ025 : Πολυμέσα**  
**Χειμερινό εξάμηνο 2017-2018**

**Προγραμματιστική άσκηση 1**

Ονοματεπώνυμο : Χρύσα Τεριζή

ΑΜ : 2553

Ημερομηνία : 17/10/2017

---

**Μέρος 1**

Κώδικας,

```
1  function [] = task1(N)
2      %Step 1
3  -    f = imread('cameraman.tif');
4  -    imagesc(f);
5  -    colormap(gray);
6
7      %Step 2
8  -    H = entropy(f);
9  -    fprintf('Entropy : %d\n', H)
10
11     %Step 3, 4
12  -    [a, b] = size(f);
13
14  -    DCTmatrix = zeros(a);
15  -    for x = 1:8:a
16  -        for y = 1:8:a
17  -            DCTmatrix(x:x+7, y:y+7) = dct2(f(x:x+7, y:y+7));
18  -        end
19  -    end
20
21     %Step 5
22  -    Q1 = N.*[16 11 10 16 24 40 51 61;
23                12 12 14 19 26 58 60 55;
24                14 13 16 24 40 57 69 56;
25                14 17 22 29 51 87 80 62;
26                18 22 37 56 68 109 103 77;
27                24 35 55 64 81 104 113 92;
28                49 64 78 87 103 121 120 101;
29                72 92 95 98 112 100 103 99];
30
```

```

31 - F_ = zeros(a);
32 - for x1 = 1:8:a
33 -     for y1 = 1:8:a
34 -         F_(x1:x1+7, y1:y1+7) = round(DCTmatrix(x1:x1+7, y1:y1+7)./Q1);
35 -     end
36 - end
37
38 %Step 6
39 F_absoluteValue = abs(F_);
40 H_ = entropy(F_absoluteValue);
41 fprintf('Entropy |F_|: %d\n', H_)
42
43 %Step 7
44 F__ = zeros(a);
45 for x2 = 1:8:a
46     for y2 = 1:8:a
47         F__(x2:x2+7, y2:y2+7) = F_(x2:x2+7, y2:y2+7).* Q1;
48     end
49 end
50
51 %Step 8
52 DCTInversematrix = zeros(a);
53 for x3 = 1:8:a
54     for y3 = 1:8:a
55         DCTInversematrix(x3:x3+7, y3:y3+7) = idct2(F__(x3:x3+7, y3:y3+7));
56     end
57 end
58
59 %Step 9
60 finalImage = uint8(DCTInversematrix);
61
62 %Step 10
63 MSE = (sum(sum((finalImage-f).*(finalImage-f)))/(a*b));
64 if(MSE == 0)
65     disp('MSE = 0')
66 else
67     PSNR = 10 * log10(255^2/MSE)
68 end
69
70 %Step 11
71 HFinalImage = entropy(finalImage);
72 fprintf('Entropy final image: %d\n', HFinalImage)
73
74 %Step 12
75 imagesc(finalImage);
76 colormap(gray);
77
78 end

```

Έχω φτιάξει μία συνάρτηση με όνομα '*task1(N)*' η οποία παίρνει σαν όρισμα τον συντελεστή του Q. Δηλαδή, όταν θέλω να το τρέξω για 2Q δεν θα αλλάξω κάτι στον κώδικα αλλά θα το τρέξω task1(2) αντίστοιχα για την περίπτωση 4Q θα τρέξω task1(4).

### Αποτελέσματα για :

#### Q = 1 \* Q1

```
>> task1(1)
Entropy : 7.009716e+00
Entropy |F_|: 6.042602e-01

PSNR =

    35.8635

Entropy final image: 7.047986e+00
```



#### Q = 2 \* Q1

```
>> task1(2)
Entropy : 7.009716e+00
Entropy |F_|: 4.509586e-01

PSNR =

    34.5233

Entropy final image: 6.574778e+00
```



$$Q = 4 * Q1$$

```
>> task1(4)
Entropy : 7.009716e+00
Entropy |F_|: 3.276775e-01

PSNR =

    33.3394

Entropy final image: 6.022211e+00
```



### Συμπεράσματα,

Βλέπουμε ότι όσο αυξάνουμε την τιμή του Q τόσο μειώνεται το PSNR και η εντροπία της εικόνας. Επίσης, αν κοιτάξουμε τις φωτογραφίες όσο αυξάνουμε το Q παρατηρούνται τα pixel της εικόνας πιο εύκολα. Κάτι το οποίο σημαίνει ότι η εικόνα μας αλλοιώνεται.

## Μέρος 2

Κώδικας,

```
1 function [] = task2(AC)
2     %AC1 = [7, 0, 0, 0, 2, 3, 0, 0, 0, 4, 0, 1, 0, 0, 0, 1, 0, 0, -1, -1, 0, 1, 0, 0, 0, 1, 0];
3     %AC2 = zeros(1,36);
4     %AC = [AC1, AC2];
5
6     [a, b] = size(AC);
7
8     disp('(LEVEL, RUN)')
9     disp('-----')
10
11     count = 0;
12     for i = 1:b
13         if (i == b)
14             disp('END OF BLOCK')
15         end
16         if (i == 1)
17             fprintf('%d, 0)\n', AC(i))
```

```

18 -         else
19 -             if(AC(i) == 0)
20 -                 count = count + 1;
21 -             else
22 -                 fprintf('%d, %d\n', AC(i), count)
23 -                 count = 0;
24 -             end
25 -         end
26 -     end
27 - end

```

Έχω φτιάξει μία συνάρτηση με όνομα *'task2(AC)'* η οποία δέχεται σαν όρισμα μία λίστα/πίνακα με τους 63 συντελεστές AC.

Φτιάχνω στο τερματικό την λίστα/πίνακα με τους συντελεστές,

```

Command Window
>> AC1 = [7, 0, 0, 0, 2, 3, 0, 0, 0, 4, 0, 1, 0, 0, 0, 1, 0, 0, -1, -1, 0, 1, 0, 0, 0, 1, 0];
>> AC2 = zeros(1,36);
>> AC = [AC1, AC2];

```

### Αποτελέσματα,

```

>> task2(AC)
(LEVEL, RUN)
-----
(7, 0)
(2, 3)
(3, 0)
(4, 3)
(1, 1)
(1, 3)
(-1, 2)
(-1, 0)
(1, 1)
(1, 3)
END OF BLOCK

```