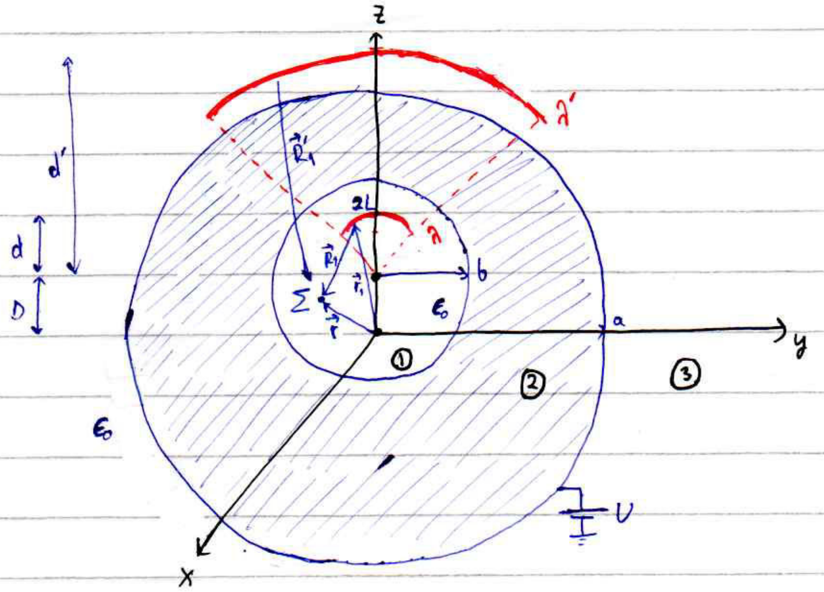
ΗΜΜΥ ΕΜΠ　　ΗΜ ΠΕΔΙΑ Β'　　ΗΒΗ ΧΑΤΖΗ
5ο ΕΞΑΜΗΝΟ　　1η Σειρά Ασκήσεων　　[redacted]

Άσκηση 6



(α) Χωρίζουμε το χώρο σε 3 περιοχές: ① στο εσωτερικό της σφαιρικής κοιλότητας
② στο εσωτερικό της αγώγιμης σφαίρας
③ έξω από τη σφαίρα

Στην περιοχή ② η σφαίρα είναι αγώγιμη σε δυναμικό U άρα $\Phi_2(x,y,z) = U$
για $(x,y,z): \sqrt{x^2+y^2+z^2} \leq a$ και $\sqrt{x^2+y^2+(z-D)^2} \geq b$

Στην περιοχή ③ απο εξίσωση Laplace έχουμε $\nabla^2\Phi_3 = 0 \xRightarrow{\text{σφαιρική συμμετρία}}$

$$\frac{1}{r^2}\frac{d}{dr}\left(r^2 \frac{d\Phi_3}{dr}\right) = 0 \Rightarrow \Phi_3(r) = -\frac{A_1}{r} + A_2$$

Συνοριακές συνθήκες: $\Phi_3(r\to\infty) = 0 \Rightarrow A_2 = 0$

$\Phi_3(r=a) = \Phi_2(r=a) = U \Rightarrow A_1 = -aU$

Άρα $\Phi_3(r) = \dfrac{aU}{r} \Rightarrow \Phi_3(x,y,z) = \dfrac{aU}{\sqrt{x^2+y^2+z^2}}$ για $\sqrt{x^2+y^2+z^2} \geq a$

Για να βρούμε το δυναμικό στην περιοχή ① αγνοούμε την αγώγιμη σφαίρα και παίρνουμε το είδωλο του τόξου σε απόσταση d' από το κέντρο της σφαιρικής κοιλότητας με γραμμική πυκνότητα φορτίου λ'
Παίρνουμε επίσης ένα φορτίο στο κέντρο τέτοιο ώστε να δώσει στο σύνορο των περιοχών ①, ② δυναμικό U αφού αλληλοεξουδετερωθούν οι συνεισφορές των τόξων

Από αρχή επαλληλίας είναι $\Phi_1(x,y,z) = \Phi_{1,\lambda}(x,y,z) + \Phi_{1,\lambda'}(x,y,z) + U$

όπου για $\sqrt{x^2+y^2+(z-D)^2} = b$ είναι $\Phi_{1,\lambda} + \Phi_{2,\lambda'} = 0$

Εξετάζουμε το $\Phi_{1,\lambda}$: είναι $\vec{r}_1 = (d\sin\theta)\hat{i}_y + (d\cos\theta + D)\hat{i}_z$

$$\vec{r} = x\hat{i}_x + y\hat{i}_y + z\hat{i}_z$$

$$\vec{R}_1 = \vec{r} - \vec{r}_1 = x\hat{i}_x + (y - d\sin\theta)\hat{i}_y + (z - d\cos\theta - D)\hat{i}_z$$

$$R_1 = |\vec{R}_1| = \left[x^2 + (y-d\sin\theta)^2 + (z-d\cos\theta-D)^2\right]^{1/2}$$

$$\Phi_{1,\lambda}(x,y,z) = \frac{1}{4\pi\varepsilon_0}\int\frac{dq}{R_1} = \frac{1}{4\pi\varepsilon_0}\int_{-L/d}^{L/d}\frac{\lambda d\, d\theta}{R_1} = \frac{\lambda d}{4\pi\varepsilon_0}\int_{-L/d}^{L/d}\frac{d\theta}{\left[x^2+(y-d\sin\theta)^2+(z-d\cos\theta-D)^2\right]^{1/2}}$$

Όμοια $\vec{R}_1' = x\hat{i}_x + (y - d'\sin\theta)\hat{i}_y + (z - d'\cos\theta - D)\hat{i}_z$

$$R_1 = \left[x^2 + (y-d'\sin\theta)^2 + (z-d'\cos\theta-D)^2\right]^{1/2}$$

$$\Phi_{1,\lambda'}(x,y,z) = \frac{\lambda' d'}{4\pi\varepsilon_0}\int_{-L/d}^{L/d}\frac{d\theta}{\left[x^2+(y-d'\sin\theta)^2+(z-d'\cos\theta-D)^2\right]^{1/2}}$$

$\Phi_{1,\lambda} + \Phi_{1,\lambda'} = 0$ για $\sqrt{x^2+y^2+(z-D)^2} = b \Rightarrow \lambda' = -\lambda\frac{d}{b}$ και $d' = \frac{b^2}{d}$

Άρα
$$\Phi(x,y,z) = \begin{cases} \Phi_{1,\lambda}(x,y,z) + \Phi_{1,\lambda'}(x,y,z) + U, & \text{για } \sqrt{x^2+y^2+(z-D)^2} < b \\ U, & \text{για } \sqrt{x^2+y^2+(z-D)^2} \geq b \text{ και } \sqrt{x^2+y^2+z^2} \leq a \\ \dfrac{aU}{\sqrt{x^2+y^2+z^2}}, & \text{για } \sqrt{x^2+y^2+z^2} > a \end{cases}$$

(β) Στην περιοχή ② $\vec{E}_2 = -\nabla\Phi_2 = 0$ αφού Φ_2 σταθερό

Στην περιοχή ③ $\vec{E}_3 = -\nabla\Phi_3 = -\frac{\partial\Phi_3}{\partial r}\hat{i}_r - \cancel{\frac{1}{r}\frac{\partial\Phi_3}{\partial\theta}\hat{i}_\theta}^{\,0} - \cancel{\frac{1}{r\sin\theta}\frac{\partial\Phi_3}{\partial\varphi}\hat{i}_\varphi}^{\,0} = \frac{aU}{r^2}\hat{i}_r = \frac{aU}{r^2}\cdot\frac{\vec{r}}{r} = \frac{aU\vec{r}}{r^3}$

$$\Rightarrow \vec{E}_3(x,y,z) = \frac{aU\,(x\hat{i}_x + y\hat{i}_y + z\hat{i}_z)}{[x^2+y^2+z^2]^{3/2}}$$

Στην περιοχή ① $d\vec{E}_1 = \frac{dq}{4\pi\varepsilon_0}\frac{\hat{i}_r}{r^2} \Rightarrow \vec{E}_1 = \frac{1}{4\pi\varepsilon_0}\int\frac{dq\,\hat{i}_r}{r^2}$ και από επαλληλία $\vec{E}_1(x,y,z) = \vec{E}_{1,\lambda} + \vec{E}_{1,\lambda'}$

$$\vec{E}_{1,\lambda} = \frac{1}{4\pi\varepsilon_0}\int\frac{dq}{R_1^2}\hat{i}_{R_1} = \frac{\lambda d}{4\pi\varepsilon_0}\int_{-L/d}^{L/d}\frac{d\theta\,\vec{R}_1}{R_1^3} = \frac{\lambda d}{4\pi\varepsilon_0}\int_{-L/d}^{L/d}\frac{d\theta\,[x\hat{i}_x + (y-d\sin\theta)\hat{i}_y + (z-d\cos\theta-D)\hat{i}_z]}{[x^2+(y-d\sin\theta)^2+(z-d\cos\theta-D)^2]^{3/2}}$$

Όμοια $$\vec{E}_{1,\lambda'} = \frac{\lambda' d'}{4\pi\varepsilon_0}\int_{-L/d}^{L/d}\frac{d\theta\,[x\hat{i}_x + (y-d'\sin\theta)\hat{i}_y + (z-d'\cos\theta-D)\hat{i}_z]}{[x^2+(y-d\sin\theta)^2+(z-d'\cos\theta-D)^2]^{3/2}}$$

Άρα $$\vec{E}(x,y,z) = \begin{cases} \vec{E}_{1,\lambda} + \vec{E}_{1,\lambda'}, & \text{για } \sqrt{x^2+y^2+(z-D)^2} < b \\ 0, & \text{για } \sqrt{x^2+y^2+(z-D)^2} > b \text{ και } \sqrt{x^2+y^2+z^2} \le a \\ \frac{aU\,(x\hat{i}_x+y\hat{i}_y+z\hat{i}_z)}{[x^2+y^2+z^2]^{3/2}}, & \text{για } \sqrt{x^2+y^2+z^2} > a \end{cases}$$

(ε) Χρησιμοποιούμε την οριακή συνθήκη $\sigma = \hat{i}_N(\cancel{D^+}^{\,0} - D^-) = -\varepsilon_0\,\hat{i}_{r_b}\,\vec{E}_1 = -\varepsilon_0\frac{\vec{r}_b}{r_b}\vec{E}_1$ και για $x=0$ (επίπεδο yz):

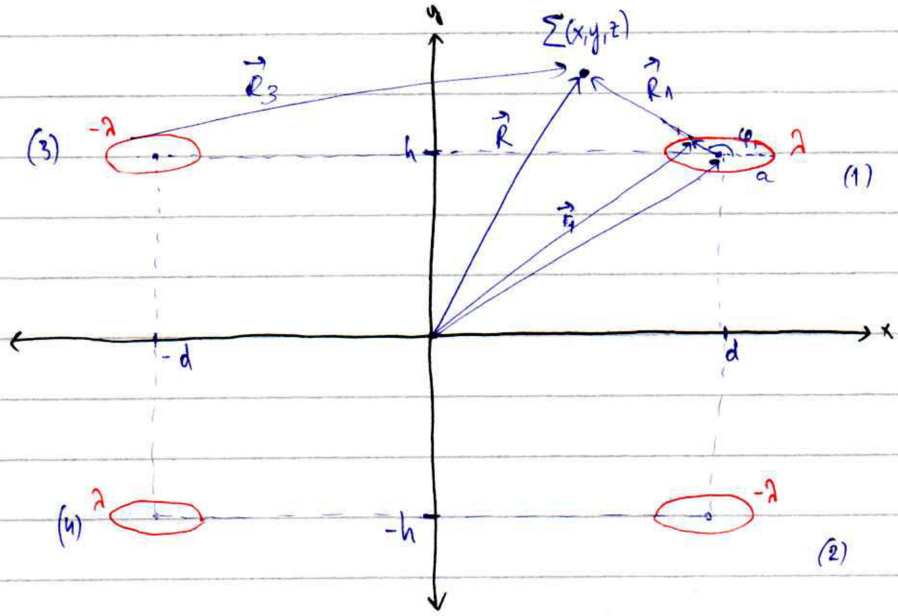
Σφαιρικές συντεταγμένες: $\sigma = \sigma(r=b,\ \theta,\ \varphi=\pi/2)$ με κέντρο το $(0,0,D)$

Καρτεσιανές: $y = b\sin\theta$, $z = b\cos\theta + D$

$\vec{r}_b = y\hat{i}_y + (z-D)\hat{i}_z$ και $r_b = \sqrt{y^2+(z-D)^2}$

Άρα $\sigma(\theta) = -\varepsilon_0\left[\frac{y}{r_b}E_{1y}(y,z) + \frac{z-D}{r_b}E_{1z}(y,z)\right]$ με $y = b\sin\theta$, $z = b\cos\theta + D$, $r_b = \sqrt{y^2+(z-D)^2}$ και $x=0$

Άσκηση 7



Το δυναμικό για $x, y \geq 0$ μπορεί να γραφεί ως επαλληλία των δυναμικών από το δακτύλιο και τα είδωλα του σχήματος (αγνοώντας τα αγώγιμα άπειρα επίπεδα)

Δηλαδή προσθέτουμε ίδιου δακτυλίους συμμετρικά στους άξονες x, y και την αρχή O, όπου οι δακτύλιοι (2) και (3) έχουν πυκνότητα $-\lambda$ και ο (4) έχει πυκνότητα λ

Με τον τρόπο αυτό το δυναμικό στους άξονες x, y για $x, y \geq 0$ διατηρείται μηδέν

Είναι $\Phi(x,y,z) = \Phi_1(x,y,z) + \Phi_2(x,y,z) + \Phi_3(x,y,z) + \Phi_4(x,y,z)$

Εξετάζουμε το Φ_1: είναι

$$\left.\begin{aligned} \vec{r}_1 &= (d + a\cos\varphi)\hat{i}_x + h\hat{i}_y + a\sin\varphi\,\hat{i}_z \\ \vec{R} &= x\hat{i}_x + y\hat{i}_y + z\hat{i}_z \\ \vec{R}_1 &= \vec{R} - \vec{r}_1 \end{aligned}\right\} \Rightarrow$$

$$\vec{R}_1 = (x - d - a\cos\varphi)\hat{i}_x + (y-h)\hat{i}_y + (z - a\sin\varphi)\hat{i}_z$$

$$R_1 = |\vec{R}_1| = \left[(x-d-a\cos\varphi)^2 + (y-h)^2 + (z-a\sin\varphi)^2\right]^{1/2} =$$

$$= \left[(x-d)^2 - 2(x-d)a\cos\varphi + a^2\cos^2\varphi + (y-h)^2 + z^2 - 2za\sin\varphi + a^2\sin^2\varphi\right]^{1/2}$$

$$= \left[(x-d)^2 + (y-h)^2 + z^2 + a^2 - 2a\left((x-d)\cos\varphi + z\sin\varphi\right)\right]^{1/2}$$

$$\Phi_1(x,y,z) = \frac{1}{4\pi\epsilon_0}\int \frac{dq_1}{R_1} = \frac{1}{4\pi\epsilon_0}\int_0^{2\pi} \frac{\lambda a\, d\varphi}{R_1} = \frac{\lambda a}{4\pi\epsilon_0}\int_0^{2\pi} d\varphi \cdot \left[(x-d)^2 + (y-h)^2 + z^2 + a^2 - 2a\left((x-d)\cos\varphi + z\sin\varphi\right)\right]^{-1/2}$$

Όμοια $\vec{R}_2 = (x-d-a\cos\varphi)\hat{i}_x + (y+h)\hat{i}_y + (z-a\sin\varphi)\hat{i}_z$

$$R_2 = \left[(x-d)^2+(y+h)^2+z^2+a^2-2a\left((x-d)\cos\varphi + z\sin\varphi\right)\right]^{1/2}$$

$$\Phi_2 = -\frac{\lambda a}{4\pi\epsilon_0}\int_0^{2\pi}\frac{d\varphi}{R_2}$$

$$R_3 = \left[(x+d)^2+(y-h)^2+z^2+a^2-2a\left((x+d)\cos\varphi+z\sin\varphi\right)\right]^{1/2}, \quad \vec{R}_3 = (x+d-a\cos\varphi)\hat{i}_x+(y-h)\hat{i}_y+(z-a\sin\varphi)\hat{i}_z$$

$$\Phi_3 = \frac{-\lambda a}{4\pi\epsilon_0}\int_0^{2\pi}\frac{d\varphi}{R_3}$$

$$R_4 = \left[(x+d)^2+(y+h)^2+z^2+a^2-2a\left((x+d)\cos\varphi+z\sin\varphi\right)\right]^{1/2}, \quad \vec{R}_4 = (x+d-a\cos\varphi)\hat{i}_x+(y+h)\hat{i}_y+(z-a\sin\varphi)\hat{i}_z$$

$$\Phi_4 = \frac{\lambda a}{4\pi\epsilon_0}\int_0^{2\pi}\frac{d\varphi}{R_4}$$

Άρα τελικά $\Phi(x,y,z) = \begin{cases} \Phi_1(x,y,z)+\Phi_2(x,y,z)+\Phi_3(x,y,z)+\Phi_4(x,y,z), & x,y>0 \\ 0, & \text{αλλού} \end{cases}$

β) $d\vec{E} = \frac{dq}{4\pi\epsilon_0}\frac{\hat{i}_R}{R^2} \Rightarrow \vec{E} = \frac{1}{4\pi\epsilon_0}\int\frac{dq\,\hat{i}_R}{R^2}$ και από επαλληλία: $\vec{E}(x,y,z) = \vec{E}_1+\vec{E}_2+\vec{E}_3+\vec{E}_4$

$$\vec{E}_1 = \frac{1}{4\pi\epsilon_0}\int\frac{dq}{R_1^2}\hat{i}_{R_1} = \frac{\lambda a}{4\pi\epsilon_0}\int_0^{2\pi}\frac{d\varphi}{R_1^2}\frac{\vec{R}_1}{R_1} = \frac{\lambda a}{4\pi\epsilon_0}\int_0^{2\pi}\frac{d\varphi\,\vec{R}_1}{R_1^3}$$

$$= \frac{\lambda a}{4\pi\epsilon_0}\int_0^{2\pi}\frac{d\varphi\left[(x-d-a\cos\varphi)\hat{i}_x+(y-h)\hat{i}_y+(z-a\sin\varphi)\hat{i}_z\right]}{\left[(x-d)^2+(y-h)^2+z^2+a^2-2a\left((x-d)\cos\varphi+z\sin\varphi\right)\right]^{3/2}}$$

Όμοια $\vec{E}_2 = \frac{-\lambda a}{4\pi\epsilon_0}\int_0^{2\pi}\frac{d\varphi\,\vec{R}_2}{R_2^3}$, $\quad \vec{E}_3 = \frac{-\lambda a}{4\pi\epsilon_0}\int_0^{2\pi}\frac{d\varphi\,R_3}{R_3^3}$, $\quad \vec{E}_4 = \frac{\lambda a}{4\pi\epsilon_0}\int_0^{2\pi}\frac{d\varphi\,R_4}{R_4^3}$

όπου $\vec{R}_i$ και R_i όπως υπολογίσαμε στο α) ερώτημα

Για $x,y\le 0$ είναι $\Phi=0$ και $\vec{E} = -\nabla\Phi = 0$

Άρα $\vec{E}(x,y,z) = \begin{cases}\vec{E}_1+\vec{E}_2+\vec{E}_3+\vec{E}_4, & x,y>0 \\ 0, & \text{αλλού}\end{cases}$

γ) Χρησιμοποιούμε την οριακή συνθήκη $\sigma = \hat{i}_n \left(D^+ - \cancel{D^2}^{\,0}\right) = \epsilon_0 E_y^+(y=0) = \frac{\epsilon_0 a}{4\pi\epsilon_0}\int_0^{2\pi} d\varphi \left[\frac{-\lambda h}{R_1^3(y=0)} + \frac{-\lambda h}{R_2^3(y=0)} + \frac{(-\lambda)(-h)}{R_3^3(y=0)} + \frac{\lambda h}{R_4^3(y=0)}\right] \Rightarrow$

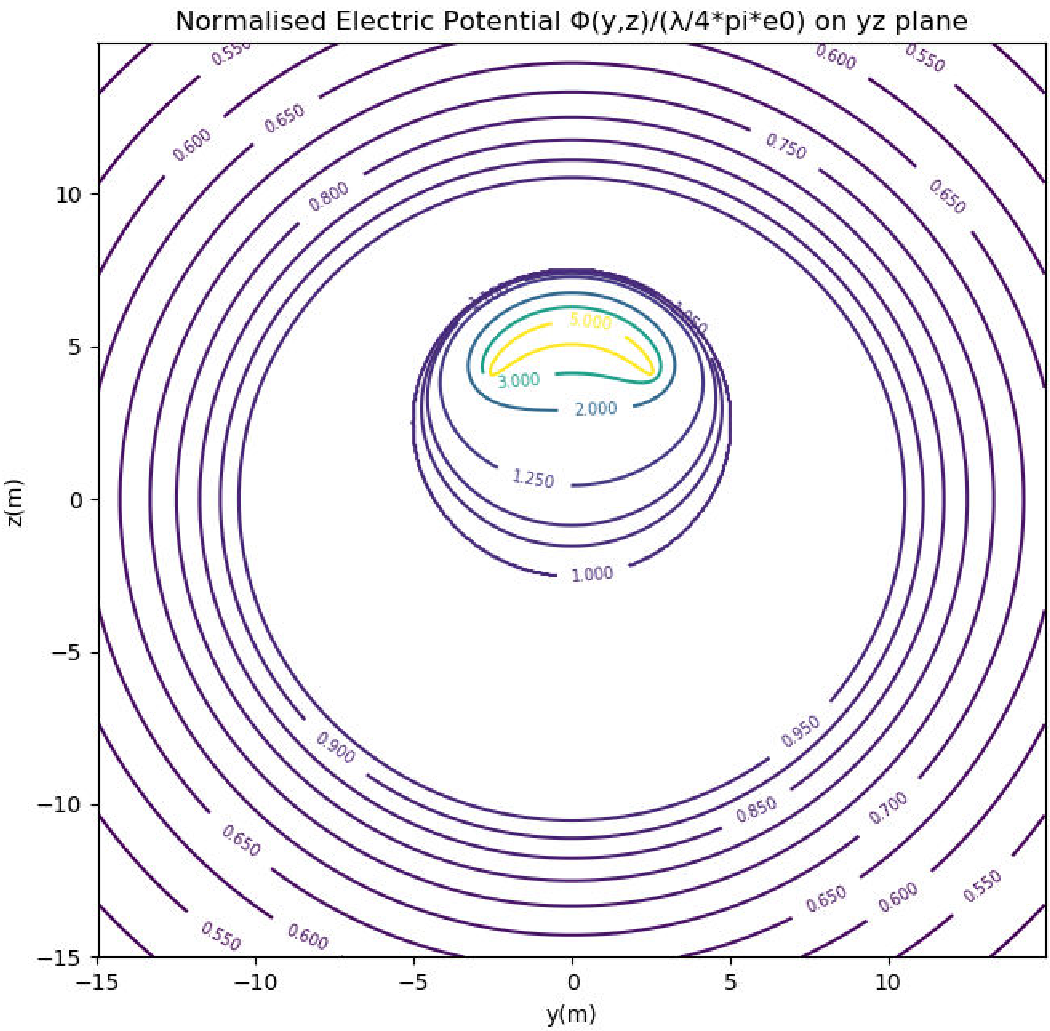
$$\sigma = \frac{\lambda a h}{4\pi}\int_0^{2\pi} d\varphi \left(-\frac{1}{R_1^3(y=0)} - \frac{1}{R_2^3(y=0)} + \frac{1}{R_3^3(y=0)} + \frac{1}{R_4^3(y=0)}\right)$$

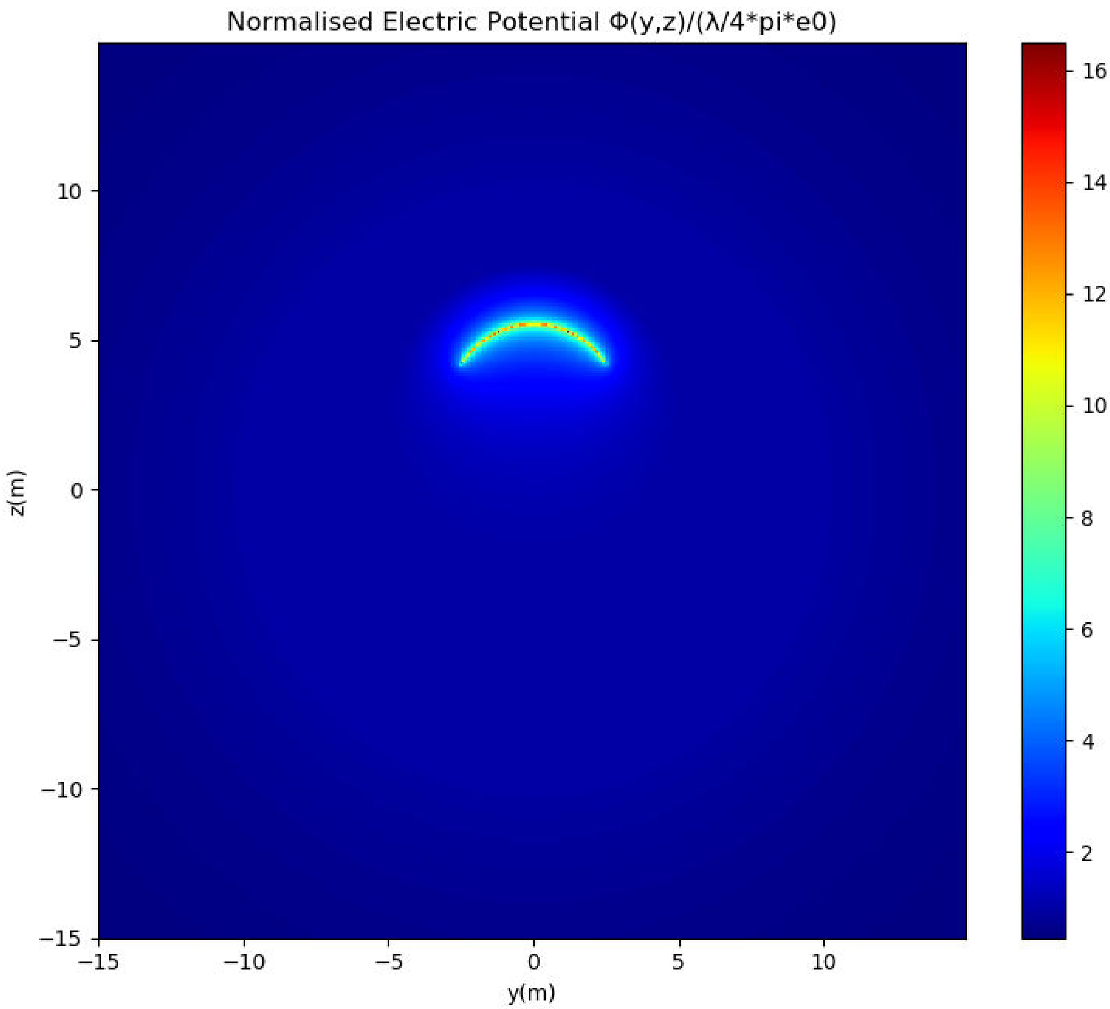
Γραφικές Παραστάσεις

έγιναν με Python 3.7

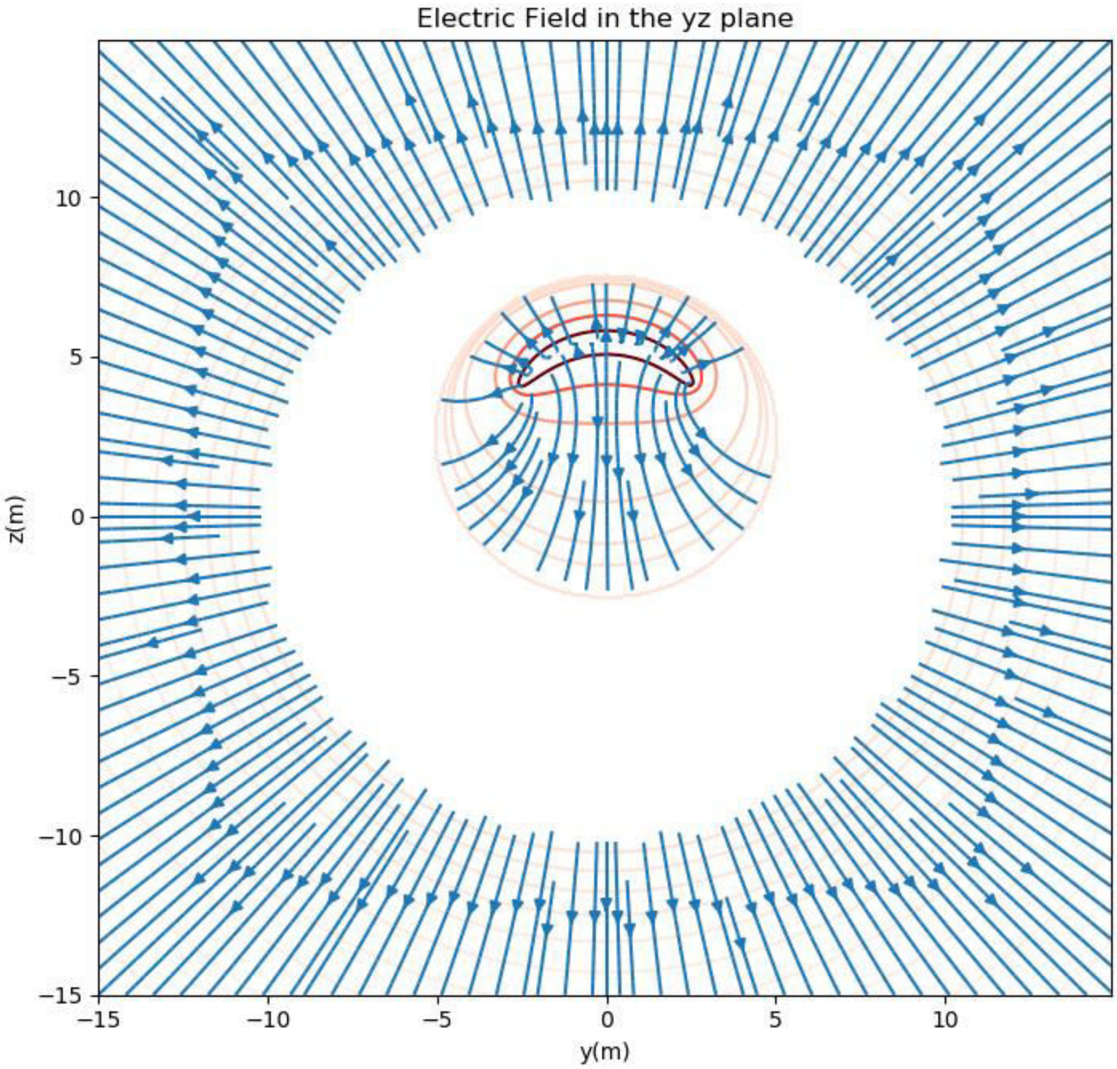
Άσκηση 6

(γ)

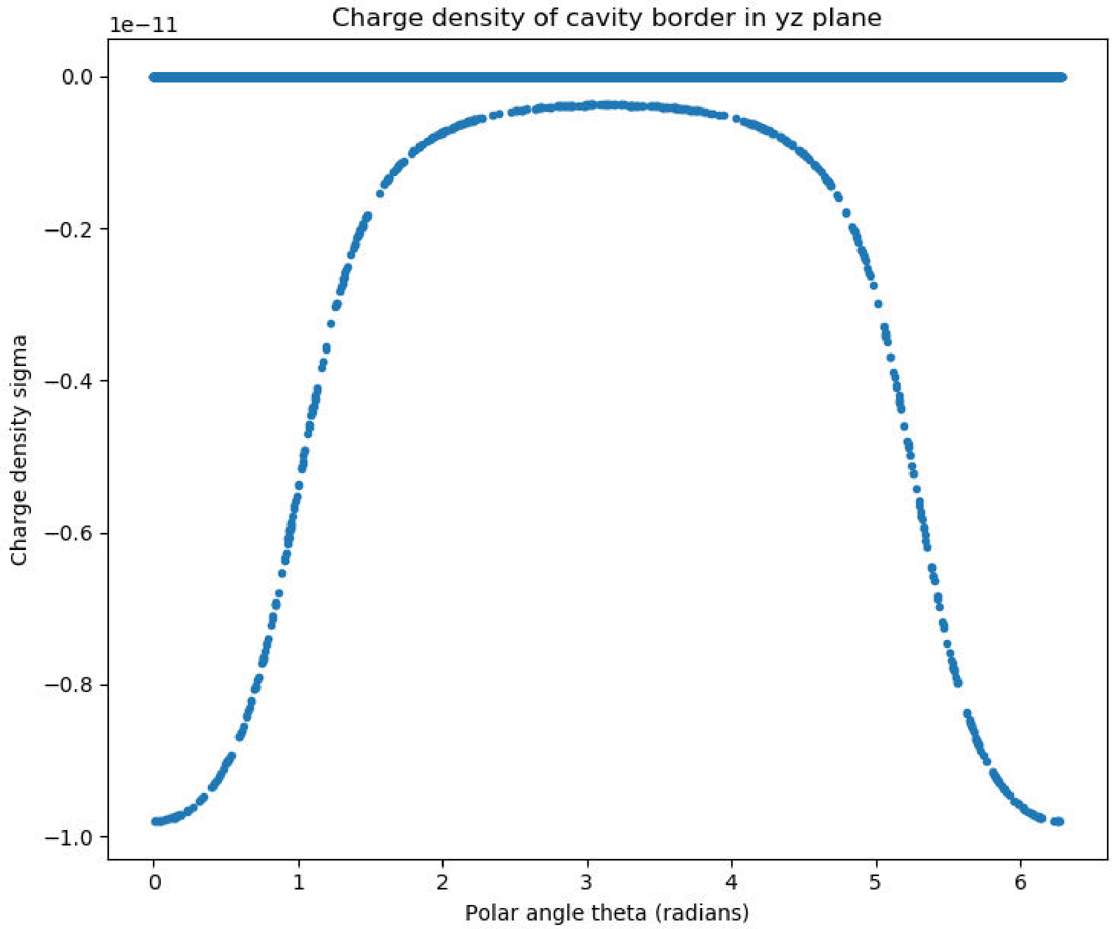



Normalised Electric Potential Φ(y,z)/(λ/4*pi*e0)
z(m)
y(m)
−15
−10
−5
0
5
10
16
14
12
10
8
6
4
2

(δ)

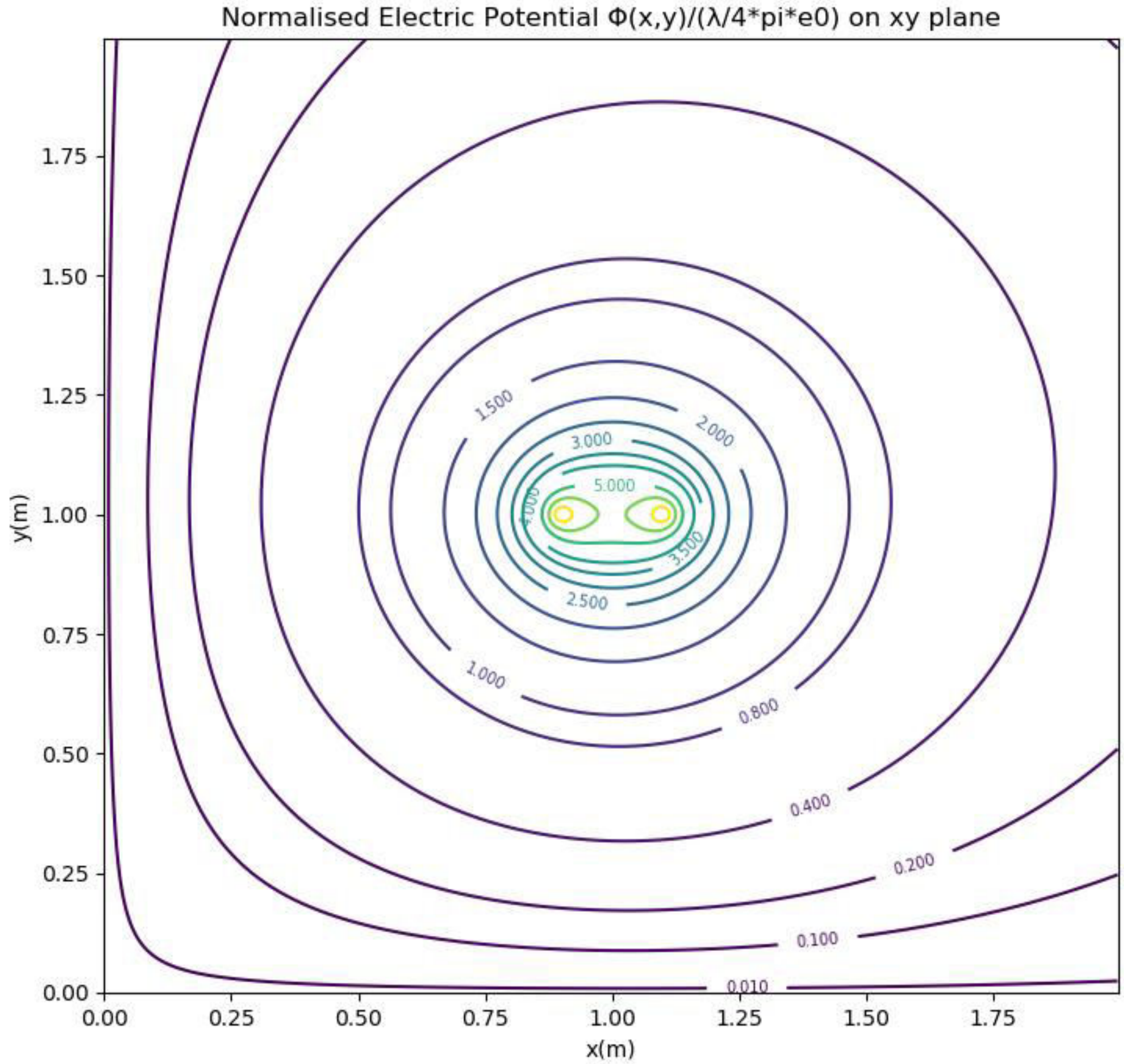


(ε)

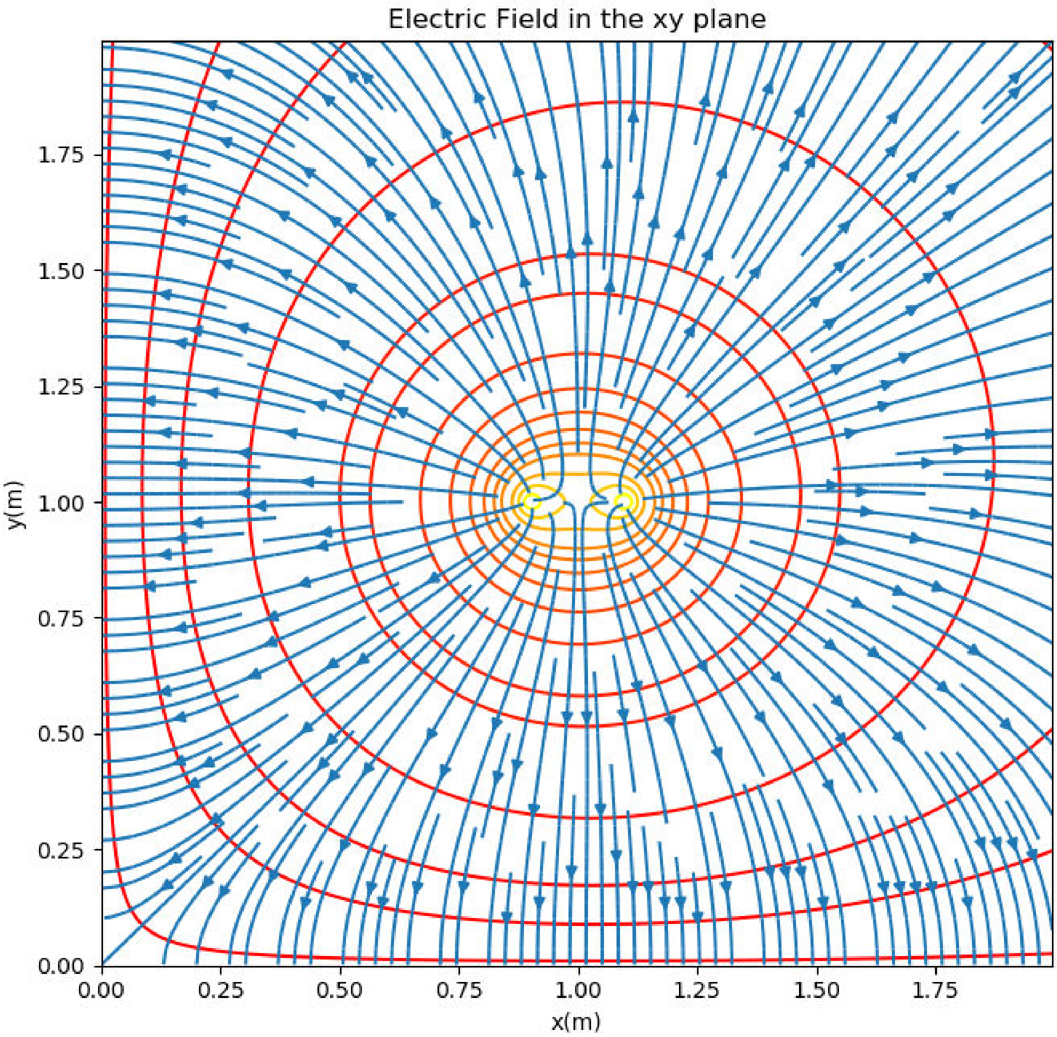

Charge density of cavity border in yz plane
1e−11
0.0
−0.2
−0.4
−0.6
−0.8
−1.0
Charge density sigma
0
1
2
3
4
5
6
Polar angle theta (radians)

Άσκηση 7

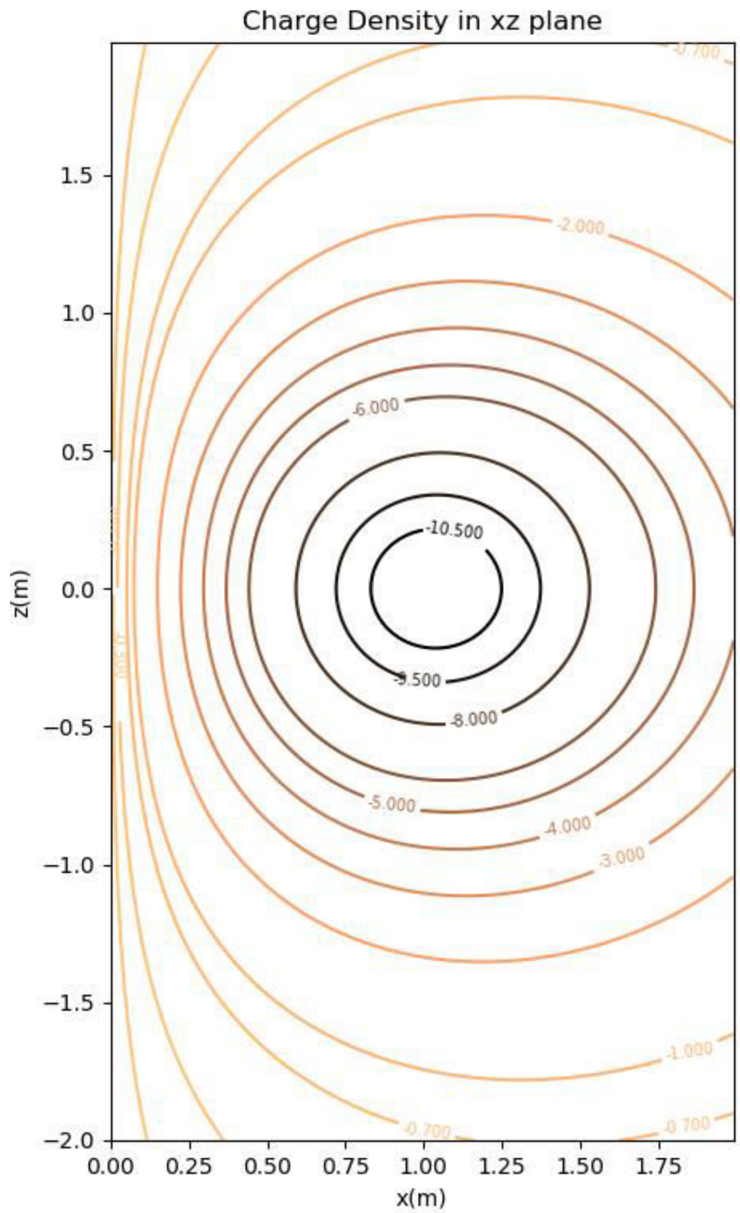
(δ)



Normalised Electric Potential Φ(x,y)/(λ/4*pi*e0) on xy plane
y(m)
x(m)
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
1.500
2.000
3.000
5.000
4.000
3.500
2.500
1.000
0.800
0.400
0.200
0.100
0.010



Electric Field in the xy plane
y(m)
x(m)
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75


Charge Density in xz plane
z(m)
x(m)
-0.700
-2.000
-6.000
-10.500
-9.500
-8.000
-5.000
-4.000
-3.000
-1.000
-0.700
-0.700
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75

Κώδικας σε Python 3.7

Άσκηση 6

```
#exercise 6
import matplotlib
import matplotlib.pyplot as plt
import matplotlib.cm as cm
import numpy as np
import math

#constants
a=10
b=5
d=3
L=3
D=2.5
V0=1
e0=8.8*(10**(-12))
levels=np.array([0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 1.0, 1.05, 1.1, 1.25, 2, 3, 5])

#Axes
ymin=-15
ymax=15
zmin=-15
zmax=15

N=400
dy=(ymax-ymin)/N
dz=(zmax-zmin)/N

yy=np.arange(ymin,ymax,dy)
zz=np.arange(zmin,zmax,dz)

Y,Z=np.meshgrid(yy,zz)



#utility function to calculate potential inside the cavity
def inside_cavity(y,z):
    thetamin=-L/d
    thetamax=L/d
    Ni=200
    dtheta=(thetamax-thetamin)/Ni
    theta=np.arange(thetamin,thetamax,dtheta)
    int1=0
    int2=0
    dd=(b**2)/d
    for i in theta:
        int1=int1+dtheta/np.sqrt((y-d*np.sin(i))**2+(z-d*np.cos(i)-D)**2)
        int2=int2+dtheta/np.sqrt((y-dd*np.sin(i))**2+(z-dd*np.cos(i)-D)**2)
    return d*int1-b*int2+V0

#electric potential function
def potential(y,z):
    r=np.sqrt(y**2+z**2)
    rT=np.sqrt(y**2+(z-D)**2)
    condlist=[rT<=b,np.logical_and(r<=a,rT>b),r>a]
    choicelist=[inside_cavity(y,z), V0, V0*a/r]
    return np.select(condlist,choicelist)

phi=potential(Y, Z)
```

```python
#utility funtions to calculate the electric field inside the cavity
def Ey_inside_cavity(y,z):
    thetamin=-L/d
    thetamax=L/d
    Ni=200
    dtheta=(thetamax-thetamin)/Ni
    theta=np.arange(thetamin,thetamax,dtheta)
    int1=0
    int2=0
    dd=(b**2)/d
    for i in theta:
        int1=int1+dtheta*(y-d*np.sin(i))/(np.sqrt((y-d*np.sin(i))**2+(z-d*np.cos(i)-D)**2))**3
        int2=int2+dtheta*(y-dd*np.sin(i))/(np.sqrt((y-dd*np.sin(i))**2+(z-dd*np.cos(i)-D)**2))**3
    return d*int1-b*int2

def Ez_inside_cavity(y,z):
    thetamin=-L/d
    thetamax=L/d
    Ni=200
    dtheta=(thetamax-thetamin)/Ni
    theta=np.arange(thetamin,thetamax,dtheta)
    int1=0
    int2=0
    dd=(b**2)/d
    for i in theta:
        int1=int1+dtheta*(z-d*np.cos(i)-D)/(np.sqrt((y-d*np.sin(i))**2+(z-d*np.cos(i)-D)**2))**3
        int2=int2+dtheta*(z-dd*np.cos(i)-D)/(np.sqrt((y-dd*np.sin(i))**2+(z-dd*np.cos(i)-D)**2))**3
    return d*int1-b*int2

#functions for electric field
def Ey_func(y,z):
    r=np.sqrt(y**2+z**2)
    rT=np.sqrt(y**2+(z-D)**2)
    condlist=[rT<b,np.logical_and(r<=a,rT>=b),r>a]
    choicelist=[Ey_inside_cavity(y,z), 0, V0*a*y/((r)**3)]
    return np.select(condlist,choicelist)

def Ez_func(y,z):
    r=np.sqrt(y**2+z**2)
    rT=np.sqrt(y**2+(z-D)**2)
    condlist=[rT<b,np.logical_and(r<=a,rT>=b),r>a]
    choicelist=[Ez_inside_cavity(y,z), 0, V0*a*z/((r)**3)]
    return np.select(condlist,choicelist)


Ey=Ey_func(Y,Z)
Ez=Ez_func(Y,Z)
```

```python
#charge density at cavity border
#the function returns zero for most points
#but plotting it over a large number of points shows the curve if zeros are ignored
def density(i):
    y=b*np.sin(i)
    z=b*np.cos(i)+D
    rb=np.sqrt(y**2+(z-D)**2)

    return -e0*(y*Ey_func(y,z)+(z-D)*Ez_func(y,z))/rb



th=np.linspace(0,2*math.pi,10000)
sigma=density(th)


#contour lines of potential
fig1, ax1 = plt.subplots()
CS = ax1.contour(Y,Z,phi,levels)
ax1.clabel(CS, inline=True, fontsize=7)
ax1.set_aspect('equal','box')
ax1.set_title('Normalised Electric Potential Φ(y,z)/(λ/4*pi*e0) on yz plane')
ax1.set_xlabel('y(m)')
ax1.set_ylabel('z(m)')

#surface plot of potential
fig2,ax2=plt.subplots()
ss=ax2.pcolormesh(Y,Z,phi,cmap=cm.jet)
ax2.set_aspect('equal','box')
ax2.set_title('Normalised Electric Potential Φ(y,z)/(λ/4*pi*e0)')
cb=fig2.colorbar(ss)
ax2.set_xlabel('y(m)')
ax2.set_ylabel('z(m)')

#streamplot of field + light coloured contour lines
fig3,ax3=plt.subplots()
CS=ax3.contour(Y,Z,phi,levels,cmap=cm.Reds)
q=ax3.streamplot(Y,Z,Ey,Ez,density=2.5)
ax3.set_aspect('equal','box')
ax3.set_title('Electric Field in the yz plane')
ax3.set_xlabel('y(m)')
ax3.set_ylabel('z(m)')

#plot of charge density
fig4, ax4 = plt.subplots()
ax4.plot(th,sigma,'.')
ax4.set_title('Charge density of cavity border in yz plane')
ax4.set_xlabel('Polar angle theta (radians)')
ax4.set_ylabel('Charge density sigma')

plt.show()
```

Άσκηση 7

```
#exercise 7
import matplotlib
import matplotlib.pyplot as plt
import matplotlib.cm as cm
import numpy as np
import math

#constants
a=0.1
d=1
h=1
e0=8.8*(10**(-12))
levels=np.array([0.01, 0.1, 0.2, 0.4, 0.8, 1.0, 1.5, 2, 2.5, 3, 3.5, 4, 5, 6, 7.5])

#Axes
xmin=0
xmax=2*d
ymin=0
ymax=2*h
zmin=-2
zmax=2

N=200
dx=(xmax-xmin)/N
dy=(ymax-ymin)/N
dz=(zmax-zmin)/N

xx=np.arange(xmin,xmax,dx)
yy=np.arange(ymin,ymax,dy)
zz=np.arange(zmin,zmax,dz)

X,Y=np.meshgrid(xx,yy)
XX,Z=np.meshgrid(xx,zz)
```

```python
#calculating potential at positive x,y
def potential_positive(x,y):
    min=0
    max=2*math.pi
    Ni=200
    di=(max-min)/Ni
    ii=np.arange(min,max,di)

    int1=0
    int2=0
    int3=0
    int4=0

    for i in ii:
        int1=int1+di/np.sqrt((x-d)**2+(y-h)**2+a**2-2*a*(x-d)*np.cos(i))
        int2=int2+di/np.sqrt((x-d)**2+(y+h)**2+a**2-2*a*(x-d)*np.cos(i))
        int3=int3+di/np.sqrt((x+d)**2+(y-h)**2+a**2-2*a*(x+d)*np.cos(i))
        int4=int4+di/np.sqrt((x+d)**2+(y+h)**2+a**2-2*a*(x+d)*np.cos(i))

    return a*int1-a*int2-a*int3+a*int4

#potential in xy plane
def potential(x,y):
    condlist=[np.logical_and(x>0,y>0),np.logical_or(x<=0,y<=0)]
    choicelist=[potential_positive(x,y),0]
    return np.select(condlist,choicelist)

phi=potential(X,Y)


#calculating electric field at positive x,y
def Ex_positive(x,y):
    min=0
    max=2*math.pi
    Ni=200
    di=(max-min)/Ni
    ii=np.arange(min,max,di)

    int1=0
    int2=0
    int3=0
    int4=0

    for i in ii:
        R1=np.sqrt((x-d)**2+(y-h)**2+a**2-2*a*(x-d)*np.cos(i))
        R2=np.sqrt((x-d)**2+(y+h)**2+a**2-2*a*(x-d)*np.cos(i))
        R3=np.sqrt((x+d)**2+(y-h)**2+a**2-2*a*(x+d)*np.cos(i))
        R4=np.sqrt((x+d)**2+(y+h)**2+a**2-2*a*(x+d)*np.cos(i))

        int1=int1+di*(x-d-a*np.cos(i))/R1**3
        int2=int2+di*(x-d-a*np.cos(i))/R2**3
        int3=int3+di*(x+d-a*np.cos(i))/R3**3
        int4=int4+di*(x+d-a*np.cos(i))/R4**3

    return a*int1-a*int2-a*int3+a*int4
```

```python
def Ey_positive(x,y):
    min=0
    max=2*math.pi
    Ni=200
    di=(max-min)/Ni
    ii=np.arange(min,max,di)

    int1=0
    int2=0
    int3=0
    int4=0

    for i in ii:
        R1=np.sqrt((x-d)**2+(y-h)**2+a**2-2*a*(x-d)*np.cos(i))
        R2=np.sqrt((x-d)**2+(y+h)**2+a**2-2*a*(x-d)*np.cos(i))
        R3=np.sqrt((x+d)**2+(y-h)**2+a**2-2*a*(x+d)*np.cos(i))
        R4=np.sqrt((x+d)**2+(y+h)**2+a**2-2*a*(x+d)*np.cos(i))

        int1=int1+di*(y-h)/R1**3
        int2=int2+di*(y+h)/R2**3
        int3=int3+di*(y-h)/R3**3
        int4=int4+di*(y+h)/R4**3

    return a*int1-a*int2-a*int3+a*int4

#electric field
def Ex_func(x,y):
    condlist=[np.logical_and(x>0,y>0),np.logical_or(x<=0,y<=0)]
    choicelist=[Ex_positive(x,y),0]
    return np.select(condlist,choicelist)

def Ey_func(x,y):
    condlist=[np.logical_and(x>0,y>0),np.logical_or(x<=0,y<=0)]
    choicelist=[Ey_positive(x,y),0]
    return np.select(condlist,choicelist)

Ex=Ex_func(X,Y)
Ey=Ey_func(X,Y)
```

```python
#calculating charge density
def density(x,z):
    min=0
    max=2*math.pi
    Ni=400
    di=(max-min)/Ni
    ii=np.arange(min,max,di)

    integ=0

    for i in ii:
        R1=np.sqrt((x-d)**2+h**2+z**2+a**2-2*a*((x-d)*np.cos(i)+z*np.sin(i)))
        R2=np.sqrt((x-d)**2+h**2+z**2+a**2-2*a*((x-d)*np.cos(i)+z*np.sin(i)))
        R3=np.sqrt((x+d)**2+h**2+z**2+a**2-2*a*((x+d)*np.cos(i)+z*np.sin(i)))
        R4=np.sqrt((x+d)**2+h**2+z**2+a**2-2*a*((x+d)*np.cos(i)+z*np.sin(i)))

        integ=integ+di*(-1/R1**3 -1/R2**3 +1/R3**3 +1/R4**3)

    return integ

sigma=density(XX,Z)



#potential contour lines
fig1, ax1 = plt.subplots()
CS = ax1.contour(X,Y,phi,levels)
ax1.clabel(CS, inline=True, fontsize=7)
ax1.set_aspect('equal','box')
ax1.set_title('Normalised Electric Potential Φ(x,y)/(λ/4*pi*e0) on xy plane')
ax1.set_xlabel('x(m)')
ax1.set_ylabel('y(m)')

#streamplot + coloured contour lines
fig2, ax2 = plt.subplots()
CS=ax2.contour(X,Y,phi,levels,cmap=cm.autumn)
q=ax2.streamplot(X,Y,Ex,Ey,density=2)
ax2.set_aspect('equal','box')
ax2.set_title('Electric Field in the xy plane')
ax2.set_xlabel('x(m)')
ax2.set_ylabel('y(m)')

#density contour lines
fig3, ax3 = plt.subplots()
levels2=np.array([-10.5,-9.5,-8,-6,-5,-4,-3,-2,-1,-0.7,-0.3,-0.1])
CS = ax3.contour(XX,Z,sigma,levels2,cmap=cm.copper)
ax3.set_aspect('equal','box')
ax3.clabel(CS, inline=True, fontsize=7)
ax3.set_title('Charge Density in xz plane')
ax3.set_xlabel('x(m)')
ax3.set_ylabel('z(m)')

plt.show()
```