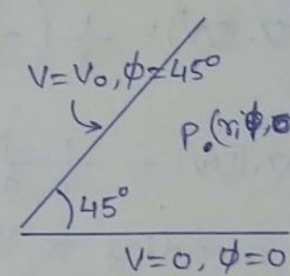
EE2010 : Engineering Electromagnetics.

Homework Assignment 3

Name :- Ganraj Achyutrao Borade.

Roll No. :- EE18BTECH11016.

1 It is given that we have a wedge capacitor with two infinite conducting plates at an angle $\phi = 45°$. The structure is invariant in the z-dir^n, and there exists an insulating gap b/w the plates.



(a) We have to find the potential between the plates by writing the general solution to Laplace equation, and then applying the boundary conditions.

→ If we confine our attention to places where there is no charge, then the poisson's equation $\left(\nabla^2 V = \frac{-\rho}{\epsilon_0}\right)$ reduces to Laplace's equation :-

$\nabla^2 V = 0$, if we written out in Cartisian coordinates,

$$\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} + \frac{\partial^2 V}{\partial z^2} = 0$$

In our case of problem, let's solve it in cylindrical coordinate system. Any point P in bet^n those plates can be written as

$P \equiv (r, \phi, 0)$,

→ $\nabla^2 V = 0$ { In between the plates as there is no charge there.)

Now $\nabla^2 V = \nabla \cdot \nabla V$

Now $\nabla V = \frac{1}{h_1}\left(\frac{\partial V}{\partial u}\hat{a}_u\right) + \frac{1}{h_2}\left(\frac{\partial V}{\partial v}\hat{a}_v\right) + \frac{1}{h_3}\left(\left(\frac{\partial V}{\partial w}\right)\hat{a}_w\right)$

In case of cylindrical coordinate system ; $h_1 = 1$, $h_2 = r$, $h_3 = 1$

$\hat{a}_u = \hat{r}$, $\hat{a}_v = \hat{\phi}$, $\hat{a}_w = \hat{z}$

$\therefore \nabla V$ (in our case) $= \left(\frac{\partial V}{\partial r}\right)\hat{r} + \frac{1}{r}\left(\frac{\partial V}{\partial \phi}\right)\hat{\phi} + \frac{1}{\cancel{z}}\underbrace{\left(\frac{\partial V}{\partial z}\right)}_{=0}\hat{z}$ (as structure is invariant in $\hat{z}$-dirn)

$$\therefore \nabla V = \left(\frac{\partial V}{\partial r}\right)\hat{r} + \frac{1}{r}\left(\frac{\partial V}{\partial \phi}\right)\hat{\phi}$$

Now $\nabla\cdot(\nabla V) = \frac{1}{h_1h_2h_3}\left(\frac{\partial(\nabla V(h_2h_3))}{\partial u} + \frac{\partial}{\partial v}(h_1h_3\nabla V) + \frac{\partial}{\partial w}(h_1h_2\nabla V)\right)$

$$\therefore \nabla\cdot(\nabla V) = \frac{1}{r}\left(\frac{\partial}{\partial r}(r\nabla V)_r + \frac{\partial}{\partial \phi}(\nabla V)_\phi + \underbrace{\frac{\partial(r(\nabla V)_z)}{\partial z}}_{=0}\right)$$

$$\therefore \nabla\cdot(\nabla V) = \frac{1}{r}\frac{\partial(r\nabla V)}{\partial r} + \frac{\partial(\nabla V)}{\partial \phi}$$

Now according to the structure, for a particular direction in which r present ($|r|$ can be of any value), the voltage along that dirn is constant.

$$\therefore \frac{\partial V}{\partial r} = 0$$

$\therefore \nabla V = \frac{1}{r}\left(\frac{\partial V}{\partial \phi}\right)\hat{\phi}$ and $\nabla\cdot(\nabla V) = \cancel{\frac{1}{r}\frac{\partial}{\partial r}\frac{r}{r}\frac{\partial V}{\partial \phi}}\, 0 + \frac{\partial(\nabla V)}{\partial \phi}$.

$$\therefore \nabla\cdot\nabla V = \nabla^2 V = \frac{\partial}{\partial \phi}\left(\frac{1}{r}\frac{\partial V}{\partial \phi}\right) = \frac{1}{r}\frac{\partial^2 V}{\partial \phi^2}$$

$\therefore \nabla^2 V = 0$ (In b/w the two plates).

$$\therefore \frac{1}{r}\frac{\partial^2 V}{\partial \phi^2} = 0 \Rightarrow \frac{\partial^2 V}{\partial \phi^2} = 0 \Rightarrow \frac{\partial V}{\partial \phi} = A$$

Hence $\boxed{V = A\phi + B} \longrightarrow$ ~~sol~~ got by solving Laplace equation

Now If we apply the boundary conditions $\Rightarrow$

At $\phi = 45°$ i.e $\phi = \frac{\pi}{4}$, $V = V_0$

and at $\phi = 0$, $V = 0 \Rightarrow 0 = 0 + B \Rightarrow \boxed{B = 0}$

and At $\phi = \frac{\pi}{4}$, $V = V_0 \Rightarrow V_0 = \left(\frac{\pi}{4}\right)A \Rightarrow \boxed{A = \frac{4V_0}{\pi}}$

Hence, $\boxed{V = \frac{4V_0\phi}{\pi}}$ Volts (In SI system) or $\boxed{\cancel{V = \frac{4V_0\phi}{\pi}}\ V}$ $\boxed{V = \frac{4V_0}{\pi}\phi\ V}$

In SI system

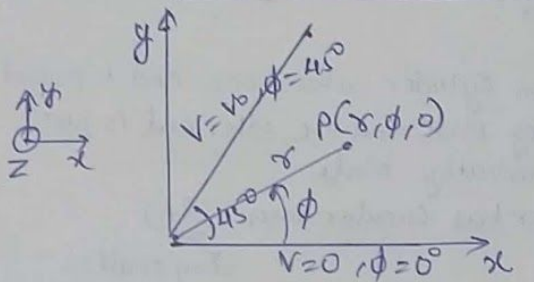
(b)

Now we know that $\vec{E}$ is a conservative field. So, it can be written as $\vec{E} = -\nabla V$

and we have calculated ∇V in part (a) :-

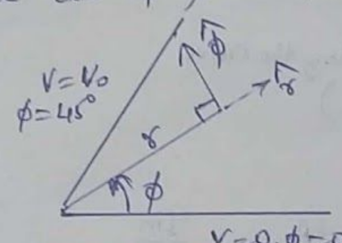
$$\nabla V = \left(\frac{\partial V}{\partial r}\right)\hat{r} + \frac{1}{r}\left(\frac{\partial V}{\partial \phi}\right)\hat{\phi} = \frac{1}{r}\left(\frac{\partial V}{\partial \phi}\right)\hat{\phi}$$

Hence $\boxed{\vec{E} = -\frac{1}{r}\left(\frac{\partial V}{\partial \phi}\right)\hat{\phi}}$



I am assuming the shown coordinate system. Here the dir^n of $\hat{r}$ is along the radial direction and the dir^n of $\hat{\phi}$ is along the tangential dir^n to $\hat{r}$ or at 90° (in the dir^n of rotation of ϕ) to $\hat{r}$.

Hence the dir^n of $\hat{\phi}$ at point P :-



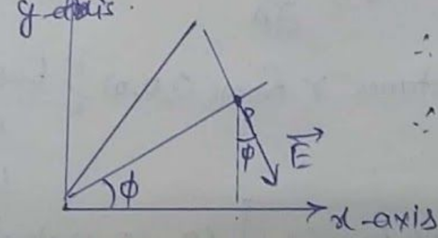
So At angle ϕ, & radial distance r the electric field $\vec{E} = \frac{-1}{r}\left(\frac{\partial V}{\partial \phi}\right)\hat{\phi}$.

$$\Rightarrow \vec{E} = \frac{-1}{r}\frac{\partial}{\partial \phi}\left(\frac{4V_0\phi}{\pi}\right)\hat{\phi}$$

$$= \frac{-1}{r} \times \frac{4V_0}{\pi}\hat{\phi}$$

Hence, $\boxed{\vec{E} = \frac{-1}{r}\frac{4V_0}{\pi}\hat{\phi}}$

Let's convert the above $\vec{E}$ into cartesian coordinate system :-



$$\therefore \vec{E} = |\vec{E}|(+\sin\phi)\hat{i} + |\vec{E}|\cos\phi(-\hat{j})$$

$$\therefore \vec{E} = \left(\frac{4V_0}{\pi r}\sin\phi\right)\hat{i} - \left(\frac{4V_0}{\pi r}\cos\phi\right)\hat{j}$$

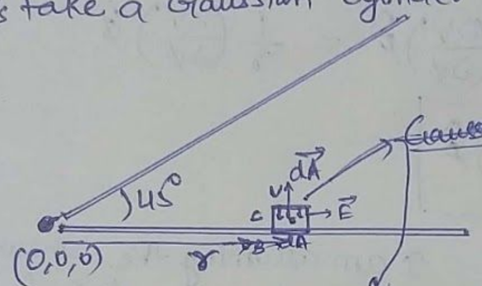
$\therefore$ $\vec{E}$ in cartesian coordinate system $= \left(\frac{4V_0}{\pi r}\sin\phi\right)\hat{i} - \left(\frac{4V_0}{\pi r}\cos\phi\right)\hat{j}$

Now we need to find surface charge density of the conductor at $\phi = 0^\circ$

Therefore, $\vec{E}$ at $\phi = 0^\circ \Rightarrow$

$$\vec{E}\Big|_{\phi=0^\circ} = \left(\frac{-4V_0}{\pi r}\right)\hat{j}$$

Now Let's take a Gaussian Cylinder like :-



~~Gaussian cylinder whose one surface is in conducting plate~~

we can assume thickness is very very small.

Gaussian cylinder whose one end is present in the conducting plate and the other end is just outside the conducting plate

(Cylinder has circular area $= dA$)

very small so we can assume σ to be const.

Hence $\oint \vec{E}\cdot d\vec{A} = \frac{q_{in}}{\epsilon_0}$ → on Gaussian cylinder

since $\vec{E}$ is $\perp^{er}$ to plate (lower), hence

$\int \vec{E_c}\cdot d\vec{A} = 0$ ($\vec{E_c}$ = Electric field along the curve and $d\vec{A} = \perp^{er}$ to $\vec{E_c}$)

$\int \vec{E_B}\cdot d\vec{A} = 0$ (As $\vec{E_B} = 0$)

$$\oint \vec{E}\cdot dA = \left(\frac{-4V_0}{\pi r}\right)\hat{j}\cdot(dA)\hat{j} = \frac{-4V_0\, dA}{\pi r} = \frac{q_{in}}{\epsilon_0}$$

$$\therefore \boxed{\frac{-4V_0\epsilon_0}{\pi r} = \frac{q_{in}}{dA}}$$

Here, q_{in} is charge on conducting plate (lower) on dA area,

$$\therefore \sigma = \frac{q_{in}}{dA}$$

$$\therefore \boxed{\sigma = \frac{-4V_0\epsilon_0}{\pi r}\ C/m^2}$$ at distance r from $(0,0,0)$ {basically $\sigma \propto \frac{1}{r}$ on lower plate}

Hence at $\phi = 0^\circ$, $\boxed{\sigma = \frac{-4V_0\epsilon_0}{\pi r}\ C/m^2}$ (In SI system)

2

(a) I have plotted the graphs for both Relative error v/s Iterations and meshgrid to show potential.

(b) Here we have

$$V = \left(\frac{4V_0}{\pi}\right)\phi \quad V \quad \text{(In SI system)}$$

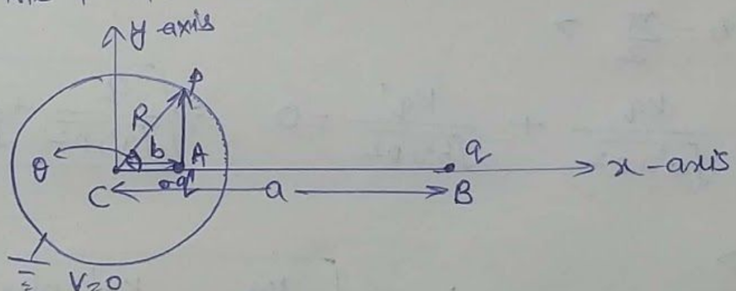
Now at $\phi = \frac{45^\circ}{2}$ i.e $\phi = \frac{\pi}{8}^c$ ($\phi = 22.5^\circ$)

then $V = \left(\frac{4V_0}{\pi}\right) \times \frac{\pi}{8} \Rightarrow V_0 = \frac{1}{2}V. \Rightarrow \boxed{V_0 = 0.5V}$

In graph also, I have plotted it.

3

It is given that a point charge q is situated at a distance a from the center of a grounded conducting sphere of radius R, & we need to find ~~Find~~ the potential outside the sphere using method of images.

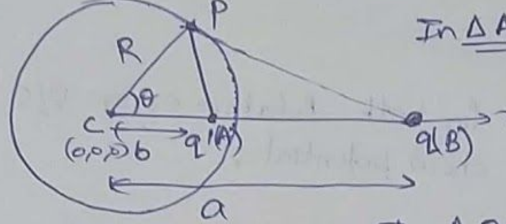


By symmetry, the image charge q' must lie somewhere on the x-axis defined by q and the center of the sphere.

Let's assume that image charge(q') is located at A and $CA = b$.

Now since the conducting sphere is grounded. So $V_{sphere} = 0$.

Now if we consider only q' and q charges and find the potential on the surface of sphere (at point P) :-



we have to find AP. and BP.

In $\Delta ACP \Rightarrow$ Now $\cos\theta = \dfrac{(b)^2+(R)^2-(AP)^2}{2bR}$

$\therefore (AP)^2 = b^2+R^2-2bR\cos\theta$

In $\Delta BCP \Rightarrow$

$$\cos\theta = \frac{a^2+R^2-(BP)^2}{2aR}$$

$$\therefore (BP)^2 = a^2+R^2-2aR\cos\theta$$

$$\therefore V_P = \frac{Kq}{(BP)} + \frac{K(q')}{(AP)} \quad \text{where } K = \frac{1}{4\pi\epsilon_0}$$

$$\boxed{\therefore V_P = \frac{Kq}{\sqrt{a^2+R^2-2aR\cos\theta}} + \frac{Kq'}{\sqrt{b^2+R^2-2bR\cos\theta}}} \rightarrow \boxed{I} \quad = 0$$

Now above formula is valid for $0 \le \theta < 2\pi$ as it should satisfy on the complete surface of sphere

$\therefore (V_P)$ at $\theta = \frac{\pi}{2} \Rightarrow$

$$V_P = \frac{Kq}{\sqrt{a^2+R^2}} + \frac{Kq'}{\sqrt{b^2+R^2}} = 0 \Rightarrow \boxed{\frac{q}{\sqrt{a^2+R^2}} + \frac{q'}{\sqrt{b^2+R^2}} = 0} \quad -(1)$$

For $\theta = 0 \Rightarrow$

$$V_P = \frac{Kq}{|a-R|} + \frac{Kq'}{|b-R|} = \underbrace{\boxed{\frac{Kq}{a-R} + \frac{Kq'}{R-b} = 0}}_{(2)} \text{ as } b \le R \text{ and } a > R$$

$\therefore$ ~~q(R-b)~~ From (2)nd eqn $\Rightarrow$

$$q^2(R-b)^2 = (q')^2(a-R)^2 \quad — (3)$$

and From (1)st eqn $\Rightarrow$

$$q^2(b^2+R^2) = (q')^2(a^2+R^2) \quad — (4)$$

Now subtracting eq^n (3) from (4) $\Rightarrow$

$$q^2(2bR) = (q')^2(2aR) \Rightarrow q^2 b = (q')^2 a$$

Now if we put above value of $(q')^2 = (q^2 b/a)$ in ~~eq^n(3)~~ eq^n(4) $\Rightarrow$

$$q^2(b^2+R^2) = q^2\left(\frac{b}{a}\right)(a^2+R^2)$$

$$\Rightarrow ab^2 + aR^2 = a^2 b + bR^2 \qquad \Rightarrow ab^2 - (a^2+R^2)b + aR^2 = 0$$

$$\Rightarrow ab^2 - a^2 b + R^2(a-b) = 0 \qquad \Rightarrow b = \frac{(a^2+R^2) \pm \sqrt{(a^2+R^2)^2 - 4a(aR^2)}}{2a}$$

$$\cancel{\Rightarrow b^2 - ab + \left(\frac{a-b}{a}\right)R^2 = 0}$$

$$\cancel{\therefore b = a \pm \sqrt{a^2-4}}$$

$$\therefore b = \frac{(a^2+R^2) \pm \sqrt{a^4+R^4+2a^2R^2-4a^2R^2}}{(2a)}$$

Since $a > R$ $\therefore |a^2-R^2| = a^2-R^2$

$$b = \frac{(a^2+R^2) \pm |a^2-R^2|}{2a}$$

$$\therefore b = \frac{(a^2+R^2) \pm (a^2-R^2)}{2a} \Rightarrow b = a \text{ or } b = R^2/a$$

at this pt. q charge is present.

Hence $\boxed{b = \frac{R^2}{a}}$

$$\therefore (q')^2 = \frac{q^2 b}{a} = q^2 \times \frac{R^2}{a^2} \Rightarrow q' = \pm \frac{qR}{a}$$

$q' = +\frac{qR}{a}$ is not acceptible as we want $\underline{V_P = 0}$

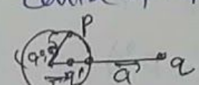
$$\therefore \boxed{q' = -q\left(R/a\right)} \text{ C (in SI system)}$$

Substituting above eq^n into $\boxed{\text{I}}$ correctly produces $V_P = 0$ for all pts. P. By uniqueness, we conclude that a point charge q located at any pt. ($\vec{a}$) outside the grounded conducting sphere of radius R produces the following electrostatic potential at any other point $\vec{r}$ outside the sphere.

8

$$\boxed{V(\vec{r}) = \frac{q}{4\pi\epsilon_0}\left\{\frac{1}{|\vec{r}-\vec{a}|} - \frac{(R/a)}{\left|\vec{r}-\left(\frac{R^2}{a^2}\right)\vec{a}\right|}\right\} \quad a, r > R}$$

By coulomb's law $\Rightarrow$ $\vec{a}$ is the ~~dist~~ position vector in which q is present.
(Assumed that centre of sphere $\equiv (0,0,0)$ origin.



$\therefore$ The potential outside the sphere using method of images $\Rightarrow$

$$V = \frac{Kq}{\cancel{4\pi\epsilon}|\vec{r}-\vec{a}|} + \frac{Kq'}{(\vec{r}-\vec{b})}$$

$$= \frac{Kq}{|\vec{r}-\vec{a}|} + \frac{K\left(-\frac{qR}{a}\right)}{\left|\vec{r}-\left(\frac{R^2}{a^2}\right)\vec{a}\right|} = \frac{q}{4\pi\epsilon_0}\left\{\frac{1}{|\vec{r}-\vec{a}|} - \frac{(R/a)}{\left|\vec{r}-\left(\frac{R^2}{a^2}\right)\vec{a}\right|}\right\}$$

$a, r > R.$

$$\therefore \boxed{V(\vec{r}) = \frac{q}{4\pi\epsilon_0}\left\{\frac{1}{|\vec{r}-\vec{a}|} - \frac{(R/a)}{\left|\vec{r}-\left(\frac{R^2}{a^2}\right)\vec{a}\right|}\right\} V \quad a, r > R \text{ (in SI system)}}$$

4

(i) $V = x^2 + y^2$

$\rightarrow$ Laplace eq^n is given by $\nabla^2 V = 0$

In cartisian Coordinate system, $\nabla^2 V = \frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} + \frac{\partial^2 V}{\partial z^2}$

$$\therefore \nabla^2 V = \frac{\partial^2 (x^2+y^2)}{\partial x^2} + \frac{\partial^2 (x^2+y^2)}{\partial y^2} + 0$$

$$\therefore \nabla^2 V = \frac{\partial^2 (x^2)}{\partial x^2} + \frac{\partial^2 (y^2)}{\partial y^2} = \frac{\partial}{\partial x}\left(\frac{\partial x^2}{\partial x}\right) + \frac{\partial}{\partial y}\left(\frac{\partial}{\partial y} y^2\right)$$

$$\therefore \nabla^2 V = \frac{\partial}{\partial x}(2x) + \frac{\partial}{\partial y}(2y) = 2 + 2 = 4$$

$$\therefore \nabla^2 V \neq 0$$

Hence $V = x^2 + y^2$ does not satisfy Laplace eq^n.

(ii) $V = x^2 - y^2$

$\rightarrow$ Laplace equation is given by $\nabla^2 V = 0$

In Cartisian Coordinate system, $\nabla^2 V = \frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} + \frac{\partial^2 V}{\partial z^2}$

$$\therefore \nabla^2 V = \frac{\partial^2 (x^2 - y^2)}{\partial x^2} + \frac{\partial^2 (x^2 - y^2)}{\partial y^2} + 0$$

$$\nabla^2 V = \frac{\partial}{\partial x}\left(\frac{\partial x^2}{\partial x}\right) + \frac{\partial}{\partial y}\left(-\frac{\partial y^2}{\partial y}\right)$$

$$\nabla^2 V = \frac{\partial}{\partial x}(2x) + \frac{\partial}{\partial y}(-2y)$$

$$\therefore \nabla^2 V = 2 - 2 = 0$$

Hence $V = x^2 - y^2$ satisfy laplace eq^n.

$\rightarrow$ General Laplace equation is $\nabla^2 V = 0$ and such a 'V' which satisfy the above condition is the solution for this laplace eq^n.

Now, the solution of Laplace eq^n can not have local maxima or minima. Extreme values must occur at end points (the boundaries). This is true because $V(x)$ is the average of $V(x+R)$ & $V(x-R)$ for any R as long as $(x+R)$ & $(x-R)$ are located in the region b/w the boundary points.

Now If we consider (i) $V = x^2 + y^2$ and plot it as a 3D-surface plot, we observe that there is a local minima at $x = y = 0$ but the Sol^n of Laplace eq^n can't have local maxima or minima. (Extreme values must occur at end pts.). Hence $V = x^2 + y^2$ is not a solution of Laplace eq^n.

~~(ii) $V = x^2 - y^2$ $\Rightarrow$ If we plot a 3D surface plot, we observe that~~
~~① Local minima or local maxima~~.

(10)

$V = x^2 + y^2$

$V_x = \frac{\partial V}{\partial x} = 2x$

$V_y = \frac{\partial V}{\partial y} = 2y$

$V_{xx} = \frac{\partial^2 V}{\partial x^2} = 2$

$V_{yy} = \frac{\partial^2 V}{\partial y^2} = 2$

$V_{xy} = \frac{\partial^2 V}{\partial x \partial y} = 0$

Now For Stationary pts, we need $V_x = 0$ and $V_y = 0$

$\therefore 2x = 0$ & $2y = 0 \Rightarrow x = 0, y = 0$.

Now we ~~need~~ need to classify it.

~~fxxf~~ $V_{xx}V_{yy} - V_{xy}^2 = (2)(2) - 0 = 4 > 0$

So it is either maxima or a minima,

But $V_{xx} = 2 > 0$ and $V_{yy} = 2 > 0$. Hence it is minimum. i.e $(0,0)$ pt is a local minima,

Hence $V = x^2 + y^2$ don't satisfy Laplace equation.

(ii) $V = x^2 - y^2$

$V_x = \frac{\partial V}{\partial x} = 2x$

$V_y = \frac{\partial V}{\partial y} = -2y$

$V_{xx} = \frac{\partial^2 V}{\partial x^2} = 2$

$V_{yy} = \frac{\partial^2 V}{\partial y^2} = -2$

$V_{xy} = \frac{\partial^2 V}{\partial x \partial y} = 0$

Now for stationary points, we need $V_x = 0$ & $V_y = 0$

$\therefore 2x = 0$ and $-2y = 0 \Rightarrow x = 0, y = 0$

Now we need to classify it.

$V_{xx}V_{yy} - V_{xy}^2 = 2(-2) - 0 = -4 < 0$

$\therefore (0,0)$ is a Saddle point,

$\therefore$ the function $V = x^2 - y^2$ has no local maxima or minima. And extreme values are occuring at end pts (Boundaries) if we see the 3D-plot

Hence $V = x^2 - y^2$ is a Sol^n for Laplace eq^n.

PROGRAMMES:
#Programmes for Question no.2:

1.

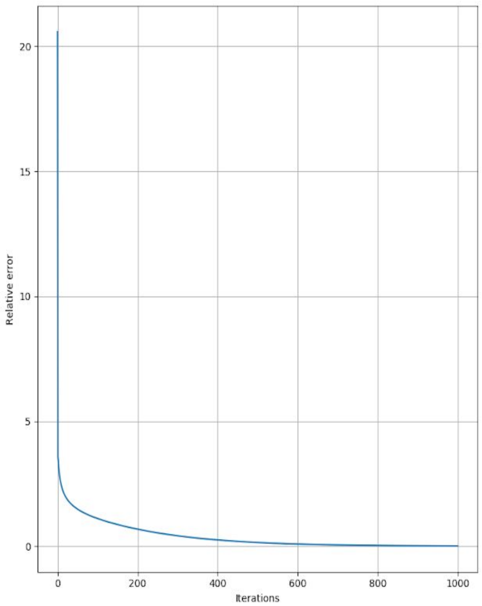
Final_2_(a)_taking_right_boundary_condition_according_to_theoretical_calculation_lenX=50.py

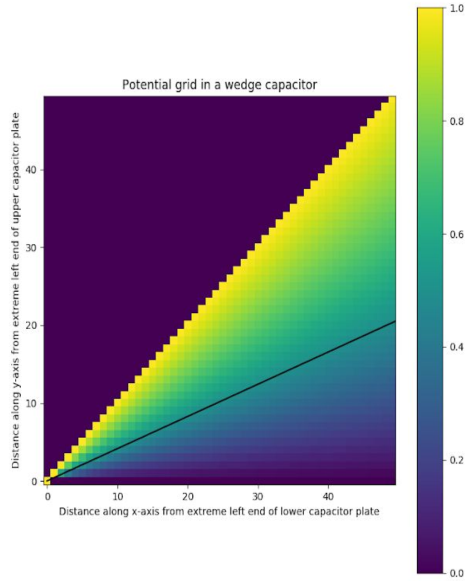
```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import cm
import math as math
from pylab import *
# Set maximum iteration
maxIter = 1000

# Set Dimension and delta
lenX = lenY = 50 #we set it rectangular
delta = 1

# Boundary condition

Tbottom = 0

Ttilted = 1
# Initial guess of interior grid
Tguess = 0

# Set colour interpolation and colour map
colorinterpolation = 100
colourMap = plt.cm.jet
x = np.linspace(0,lenX-0.5,100)
# Set meshgrid
X, Y = np.meshgrid(np.arange(0, lenX), np.arange(0, lenY))

# Set array size and set the interior value with Tguess
T = np.empty((lenX, lenY))
T.fill(Tguess)
```

```
# Set Boundary condition

T[:1, :] = Tbottom

for v in range(0,lenY):
      T[v,(lenX-1):] = (4/np.pi)*(math.atan(v/(lenX)))

print("The above formulations for the boundary condition on the right boundary of the
meshgrid is done because if we take any boundary fixed value at right boundary,then
the plot which will be obtained will satisfy the theoretical calculation only upto a
particular point in the meshgrid.After that point the plot becomes slightly distorted
because we actually don't know the right boundary condition.So to avoid this
disruption we are formulating the value at each point on right boundary according to
the theoretical calculation.")

print("**********************************************************************
**********************")

T[:(lenX-1), :(lenY-1)] = 0
for t in range(lenX):
      T[lenX-1-t,lenX-1-t] = 1

# Iteration (We assume that the iteration is convergence in maxIter = 1000)
s =0
values = np.zeros(maxIter)
for iteration in range(0, maxIter):
   for i in range(1, lenX-1):
      for j in range(1, i):
         T[lenX-1-i, lenX-1-j] = 0.25 * (T[lenX-1-i+1][lenX-1-j] +
T[lenX-1-i-1][lenX-1-j] + T[lenX-1-i][lenX-1-j+1] + T[lenX-1-i][lenX-1-j-1])
   for b in range(1,lenX-1):
      for c in range(1,lenY-1):
            s = s + (T[b,c])**2

   values[iteration] = s
   s=0
err = np.zeros(maxIter)
for i in range(maxIter):
      err[i] = (np.abs(values[i-1]-values[i]))**0.5
      #print(err[i])
print("Iteration finished")
itr = np.linspace(0,maxIter-1,maxIter)
subplot(1,2,1)
```

```
plt.plot(itr,err)
plt.grid()
plt.xlabel("Iterations")
plt.ylabel("Relative error")
subplot(1,2,2)
plt.title("Potential grid in a wedge capacitor")
#plt.contourf(X, Y, T, colorinterpolation, cmap=colourMap)
plt.imshow(T, origin='higher', interpolation=None,cmap='viridis')
y = ((2**(0.5)) - 1)*x
plt.plot(x,y,'k')
print(T)

print("****************************************")
print("From the graph of Relative error vs Iterations ,we observe that after 600 iteratons,the value of Relative error converges.So minimum 600 iterations in this case are required to get reasonable convergence.")
# Set Colorbar
plt.colorbar()
plt.xlabel('Distance along x-axis from extreme left end of lower capacitor plate')
plt.ylabel('Distance along y-axis from extreme left end of upper capacitor plate')
# Show the result in the plot window
plt.show()
```

<u>Diagram got after running the above code :</u>





2.

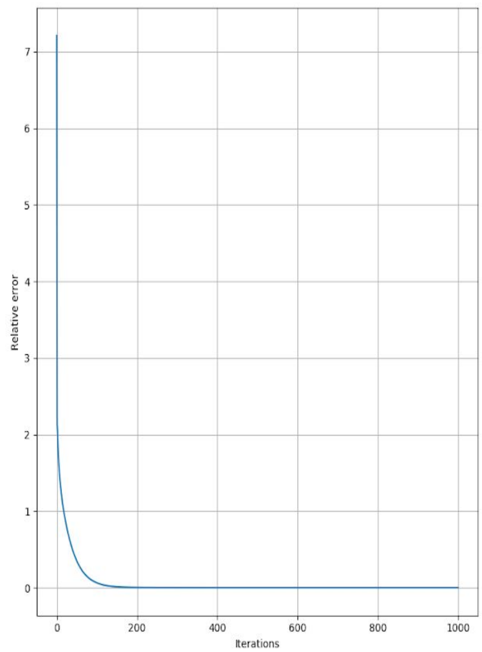
Final_2_(a)_taking_right_boundary_condition_according_to_theoretical_calculation_with_lenX=20.py

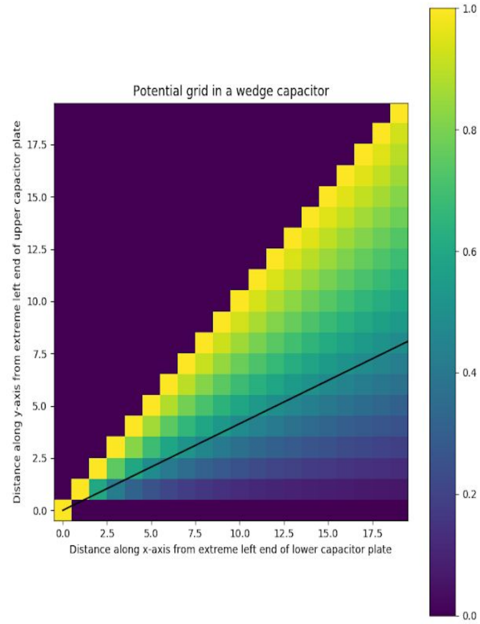
```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import cm
import math as math
from pylab import *
# Set maximum iteration
maxIter = 1000

# Set Dimension and delta
lenX = lenY = 20 #we set it rectangular
delta = 1

# Boundary condition

Tbottom = 0

Ttilted = 1
```

```
# Initial guess of interior grid
Tguess = 0

# Set colour interpolation and colour map
colorinterpolation = 100
colourMap = plt.cm.jet
x = np.linspace(0,lenX-0.5,100)
# Set meshgrid
X, Y = np.meshgrid(np.arange(0, lenX), np.arange(0, lenY))

# Set array size and set the interior value with Tguess
T = np.empty((lenX, lenY))
T.fill(Tguess)

# Set Boundary condition

T[:1, :] = Tbottom

for v in range(0,lenY):
      T[v,(lenX-1):] = (4/np.pi)*(math.atan(v/(lenX)))

print("The above formulations for the boundary condition on the right boundary of the
meshgrid is done because if we take any boundary fixed value at right boundary,then
the plot which will be obtained will satisfy the theoretical calculation only upto a
particular point in the meshgrid.After that point the plot becomes slightly distorted
because we actually don't know the right boundary condition.So to avoid this
disruption we are formulating the value at each point on right boundary according to
the theoretical calculation.")

print("****************************************************************************
************************")

T[:(lenX-1), :(lenY-1)] = 0
for t in range(lenX):
      T[lenX-1-t,lenX-1-t] = 1

# Iteration (We assume that the iteration is convergence in maxIter = 1000)
s =0
values = np.zeros(maxIter)
for iteration in range(0, maxIter):
    for i in range(1, lenX-1):
        for j in range(1, i):
```

```
            T[lenX-1-i, lenX-1-j] = 0.25 * (T[lenX-1-i+1][lenX-1-j] + T[lenX-1-i-1][lenX-1-j] + T[lenX-1-i][lenX-1-j+1] + T[lenX-1-i][lenX-1-j-1])
    for b in range(1,lenX-1):
        for c in range(1,lenY-1):
                s = s + (T[b,c])**2

    values[iteration] = s
    s=0
err = np.zeros(maxIter)
for i in range(maxIter):
        err[i] = (np.abs(values[i-1]-values[i]))**0.5
        #print(err[i])
print("Iteration finished")
itr = np.linspace(0,maxIter-1,maxIter)
subplot(1,2,1)
plt.plot(itr,err)
plt.grid()
plt.xlabel("Iterations")
plt.ylabel("Relative error")
subplot(1,2,2)
plt.title("Potential grid in a wedge capacitor")
#plt.contourf(X, Y, T, colorinterpolation, cmap=colourMap)
plt.imshow(T, origin='higher', interpolation=None,cmap='viridis')
y = ((2**(0.5)) - 1)*x
plt.plot(x,y,'k')
print(T)

print("****************************************")
print("From the graph of Relative error vs Iterations ,we observe that after 200 iteratons,the value of Relative error converges.So minimum 200 iterations in this case are required to get reasonable convergence.")
# Set Colorbar
plt.colorbar()
plt.xlabel('Distance along x-axis from extreme left end of lower capacitor plate')
plt.ylabel('Distance along y-axis from extreme left end of upper capacitor plate')
# Show the result in the plot window
plt.show()
```

<u>Diagram got after running the above code :</u>





3.

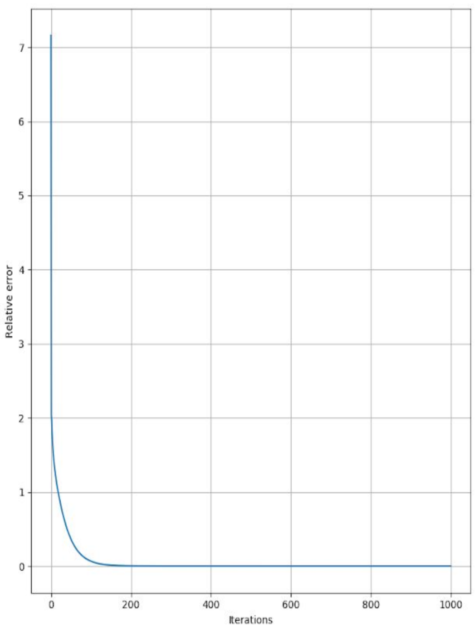
2_(a)_assuming_right_boundary_condition_tobe_ofValue_0.5_with_lenX=20.py

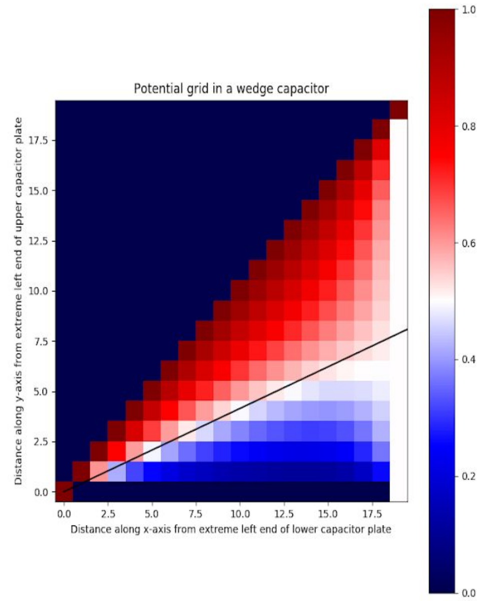
```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import cm
import math as math
from pylab import *
# Set maximum iteration
fig, ax = plt.subplots()
maxIter = 1000

# Set Dimension and delta
lenX = lenY = 20 #we set it rectangular
delta = 1

# Boundary condition
```

```
Tbottom = 0

Ttilted = 1
# Initial guess of interior grid
Tguess = 0

# Set colour interpolation and colour map
colorinterpolation = 100
colourMap = plt.cm.jet
x = np.linspace(0,lenX-0.5,100)
# Set meshgrid
X, Y = np.meshgrid(np.arange(0, lenX), np.arange(0, lenY))

# Set array size and set the interior value with Tguess
T = np.empty((lenX, lenY))
T.fill(Tguess)

# Set Boundary condition

T[:1, :] = Tbottom

# for v in range(0,lenY):
#       T[v,(lenX-1):] = (4/np.pi)*(math.atan(v/(lenX)))
T[:,(lenX-1):] = 0.5
print("We are taking the right boundary condition to be equal to of value 0.5 units.")
print("******************************************************************")

T[:(lenX-1), :(lenY-1)] = 0
for t in range(lenX):
      T[lenX-1-t,lenX-1-t] = 1

# Iteration (We assume that the iteration is convergence in maxIter = 1000)
s =0
values = np.zeros(maxIter)
print("Please wait for a moment")
for iteration in range(0, maxIter):
    for i in range(1, lenX-1):
        for j in range(1, i):
            T[lenX-1-i, lenX-1-j] = 0.25 * (T[lenX-1-i+1][lenX-1-j] +
T[lenX-1-i-1][lenX-1-j] + T[lenX-1-i][lenX-1-j+1] + T[lenX-1-i][lenX-1-j-1])
    for b in range(1,lenX-1):
       for c in range(1,lenY-1):
```

```
            s = s + (T[b,c])**2

    values[iteration] = s
    s=0
err = np.zeros(maxIter)
for i in range(maxIter):
        err[i] = (np.abs(values[i-1]-values[i]))**0.5
        #print(err[i])
print("Iteration finished")
itr = np.linspace(0,maxIter-1,maxIter)
subplot(1,2,1)
plt.plot(itr,err)
plt.grid()
plt.xlabel("Iterations")
plt.ylabel("Relative error")
subplot(1,2,2)
plt.title("Potential grid in a wedge capacitor")
#plt.contourf(X, Y, T, colorinterpolation, cmap=colourMap)
plt.imshow(T, origin='higher', interpolation=None,cmap='seismic')
y = ((2**(0.5)) - 1)*x
plt.plot(x,y,'k')
print(T)

print("****************************************")
print("From the graph of Relative error vs Iterations ,we observe that after 200
iteratons,the value of Relative error converges.So minimum 200 iterations in this case
are required to get reasonable convergence.")
# Set Colorbar
plt.colorbar()
plt.xlabel('Distance along x-axis from extreme left end of lower capacitor plate')
plt.ylabel('Distance along y-axis from extreme left end of upper capacitor plate')
# Show the result in the plot window
plt.show()
```

Diagram got after running the above code :





4.

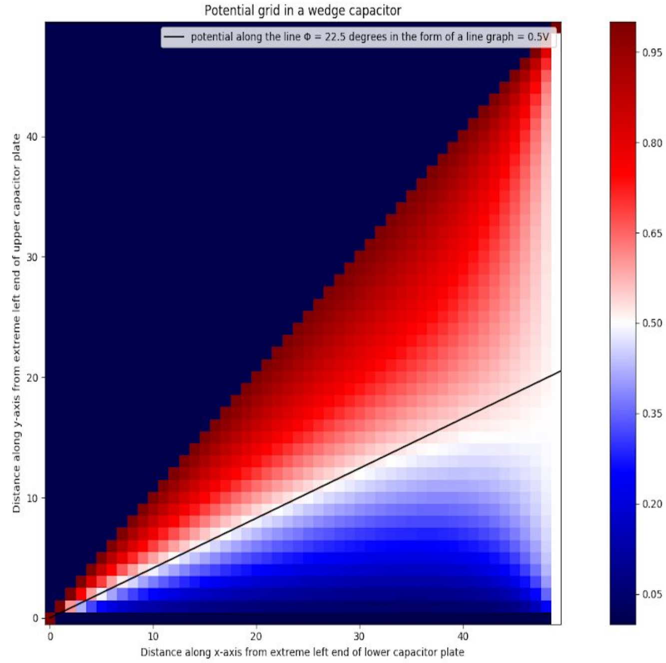
2_(b).py

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import cm
import math as math
from pylab import *
# Set maximum iteration
maxIter = 1000

# Set Dimension and delta
lenX = lenY = 50 #we set it rectangular
delta = 1

# Boundary condition

Tbottom = 0
```

```
Ttilted = 1
# Initial guess of interior grid
Tguess = 0

# Set colour interpolation and colour map
colorinterpolation = 100
colourMap = plt.cm.jet
x = np.linspace(0,lenX-0.5,100)
# Set meshgrid
X, Y = np.meshgrid(np.arange(0, lenX), np.arange(0, lenY))

# Set array size and set the interior value with Tguess
T = np.empty((lenX, lenY))
T.fill(Tguess)

# Set Boundary condition

T[:1, :] = Tbottom

T[:,(lenX-1):] = 0.5
print("We are taking the right boundary condition to be equal to of value 0.5 units.")

print("***************************************************************************
************************")

T[:(lenX-1), :(lenY-1)] = 0
for t in range(lenX):
	T[lenX-1-t,lenX-1-t] = 1

# Iteration (We assume that the iteration is convergence in maxIter = 1000)

for iteration in range(0, maxIter):
    for i in range(1, lenX-1):
        for j in range(1, i):
            T[lenX-1-i, lenX-1-j] = 0.25 * (T[lenX-1-i+1][lenX-1-j] +
T[lenX-1-i-1][lenX-1-j] + T[lenX-1-i][lenX-1-j+1] + T[lenX-1-i][lenX-1-j-1])

print("Iteration finished")
itr = np.linspace(0,maxIter-1,maxIter)
plt.title("Potential grid in a wedge capacitor")
#plt.contourf(X, Y, T, colorinterpolation, cmap=colourMap)
plt.imshow(T, origin='higher', interpolation=None,cmap='seismic')
y = ((2**(0.5)) - 1)*x
```

```
plt.plot(x,y,'k',label= 'potential along the line Φ = 22.5 degrees in the form of a line
graph = 0.5V ')
print('Since this line is passing through white boxes sufficiently from start and white
box corresponds to 0.5V .Hence the potential along this line is 0.5V ')
print('*****************************************************************************
********************')
print(T)

print("******************************************")
print("From the graph of Relative error vs Iterations ,we observe that after 600
iteratons,the value of Relative error converges.So minimum 600 iterations in this case
are required to get reasonable convergence.")
# Set Colorbar
v = np.linspace(-.1, 2.0, 15, endpoint=True)
plt.colorbar(ticks=v)
plt.xlabel('Distance along x-axis from extreme left end of lower capacitor plate')
plt.ylabel('Distance along y-axis from extreme left end of upper capacitor plate')
# Show the result in the plot window

plt.legend()
plt.show()
```

<u>Diagram got after running the above code :</u>



#Programmes for Question no.4:

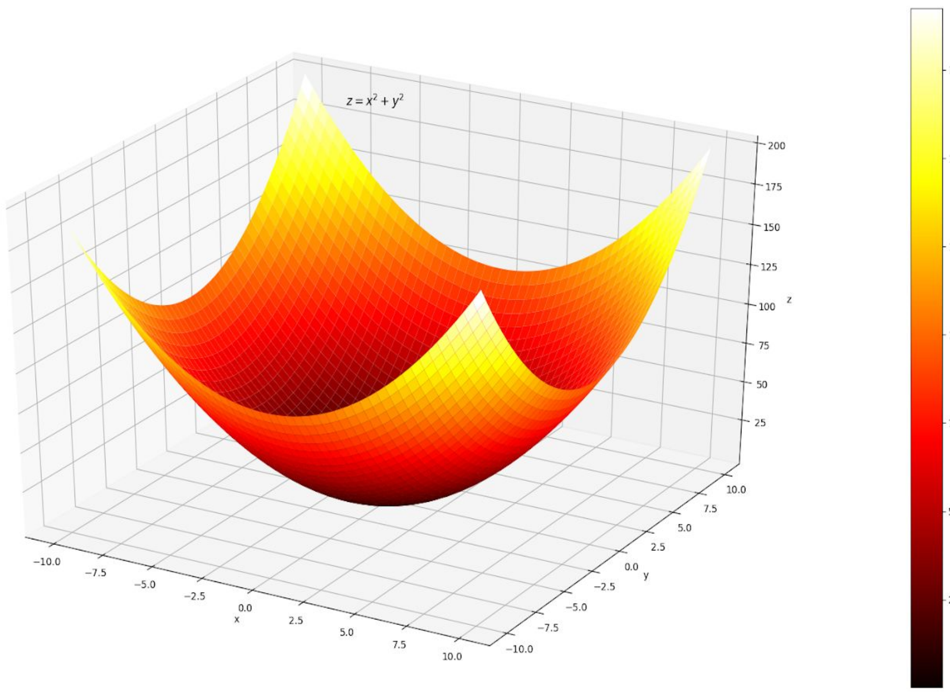
1.

4_(1).py

```
import numpy as np
from mpl_toolkits import mplot3d
import matplotlib.pyplot as plt

from matplotlib import cm
fig = plt.figure(num=1)
fig.clf()
ax = fig.add_subplot(1, 1, 1, projection='3d')

(x, y) = np.meshgrid(np.linspace(-10, 10, 1000),
                     np.linspace(-10, 10, 1000))
z =(x**2+y**2)

p = ax.plot_surface(x, y, z, cmap=cm.hot)
ax.set(xlabel='x', ylabel='y', zlabel='z',
```

```
        title='Rectilinear Grid')
fig.colorbar(p)
fig.tight_layout()
plt.title("$z = x^2 + y^2$")
plt.show()
```

__Diagram got after running the above code :__



2.

4_(2).py

```
import numpy as np
from mpl_toolkits import mplot3d
import matplotlib.pyplot as plt
from matplotlib import cm
fig = plt.figure(num=1)
fig.clf()
ax = fig.add_subplot(1, 1, 1, projection='3d')

(x, y) = np.meshgrid(np.linspace(-10, 10, 1000),
                     np.linspace(-10, 10, 1000))
```

```
z =(x**2-y**2)

p = ax.plot_surface(x, y, z, cmap=cm.hot)
ax.set(xlabel='x', ylabel='y', zlabel='z',
      title='Rectilinear Grid')
fig.colorbar(p)
fig.tight_layout()
plt.title("$z = x^2 - y^2$")
plt.show()
```

<u>Diagram got after running the above code :</u>