In [1]:
```python
# Requisites
import numpy as np
import scipy.constants as const

# Constants
e0= const.epsilon_0
e = const.elementary_charge
k = 1/(4*np.pi*e0)


def mag_hat(vector):
    # Calculate magnitude and unit vectors
    magnitude = np.linalg.norm(vector)
    hat = vector/magnitude
    return magnitude, hat

def e_calc(charge, rmag, rhat):
    # Calculate electric field, EQN: (kq/|r|^2)*rhat
    e_field = (k*charge/rmag**2)*rhat
    return e_field

def eforce_calc(charge, efield):
    force = charge*efield
    return force
```

In [ ]:
```python
q10_1 = 2e-6
q10_2 = 7e-6
q10_3 = -5e-6

loc10_q1 = np.array([0,0.04,0])
loc10_q2 = np.array([0,0,0])
loc10_q3 = np.array([0.03,0,0])

r10_21 = loc10_q1
r10_31 = loc10_q1 - loc10_q3

r10_21_mag, r10_21_hat = mag_hat(r10_21)
r10_31_mag, r10_31_hat = mag_hat(r10_31)

e10_21 = e_calc(q10_2, r10_21_mag, r10_21_hat)
e10_31 = e_calc(q10_3, r10_31_mag, r10_31_hat)

e10_2131 = e10_21 + e10_31
#print(e10_2131)

f10_2131 = q10_1 * e10_2131
#print(f10_2131)

loc10_a = np.array([0.03,0.04,0])

r10_q1a = loc10_q3
r10_q2a = loc10_a
r10_q3a = loc10_q1
```

```
r_q1a_mag, r_q1a_hat = mag_hat(r10_q1a)
r_q2a_mag, r_q2a_hat = mag_hat(r10_q2a)
r_q3a_mag, r_q3a_hat = mag_hat(r10_q3a)

e10_1a = e_calc(q10_1, r_q1a_mag, r_q1a_hat)
e10_2a = e_calc(q10_2, r_q2a_mag, r_q2a_hat)
e10_3a = e_calc(q10_3, r_q3a_mag, r_q3a_hat)

e10_anet = e10_1a + e10_2a + e10_3a

print("E net: ",e10_anet)
m10 = 6.646e-27

f10_qe = 2*e*e10_anet
f10_acc = f10_qe/m10

print("F net: ", f10_acc)

asdasdadsadasdasd
```

```
E net:  [35071424.3271347  -7953983.33615114        0.        ]
F net:  [ 1.69096047e+15 -3.83499436e+14  0.00000000e+00]
```

In [ ]: