To help you practice for the coding I made a stump code similar to what might be seen on your test. Things you may have to do include:

1. Define constants, POI position, etc
2. Compute $\vec{r}$
    a. *From source* to POI for electric field
    b. *From charge causing force* to charge experiencing force for Coulomb's law
3. Compute Coulomb force of one charge on another
4. Compute electric field due to point charges
5. Animate motion using the Euler-Cromer Method (ECM)
   ECM can be implemented using momentum or velocity as shown below.

```
GlowScript 3.1 VPython

#define any constants you need!!!

frog = sphere()
frog.pos = vec(4,0,0)               #in m
frog.velocity = vec(0,0,0)          #in m/s
frog.acceleration = vec(0,0,0)      #in m/2^2
frog.charge = 1e-5                  #in C
frog.mass = 1e3                    #in kg

cow = sphere()
cow.pos = vec(4,0,0)               #in m
cow.velocity = vec(0,0,0)          #in m/s
cow.acceleration = vec(0,0,0)      #in m/2^2
cow.charge = -1e-5                  #in C
cow.mass = 1e-3                    #in kg

#what would you do to compute net electric field
#at POI caused by charges frog & cow?

t=0
dt=0.1
while t<10:
    rate(1/dt)
    #what would you do to animate motion of cow
    #for force that frog exerts on cow
```

$\vec{F}_{NET} \rightarrow \vec{a} \rightarrow \vec{v} \rightarrow \vec{r}$	$\vec{F}_{NET} \rightarrow \vec{p} \rightarrow \vec{r}$
• Compute $\vec{F}_{NET}$. • Use $\vec{a} = \frac{\vec{F}_{NET}}{m}$ (Newton's 2nd Law) • Use $\Delta\vec{v} = \vec{a}\Delta t$ ($\vec{a} = \frac{\Delta\vec{v}}{\Delta t}$) Equivalent to $\vec{v}_f = \vec{v}_i + \vec{a}t$ In code `v += a*dt` • Use $\Delta\vec{r} = \vec{v}\Delta t$ ($\vec{v} = \frac{\Delta\vec{r}}{\Delta t}$) Equivalent to $\vec{r}_f = \vec{r}_i + \vec{v}t$ In code `ball.pos += v*dt` • Increment time using `t += dt`	• Compute $\vec{F}_{NET}$. • Use $\Delta\vec{p} = \vec{F}\Delta t$ ($\vec{F} = \frac{\Delta\vec{p}}{\Delta t}$) This is a variation of Newton's 2nd Law Equivalent to $\vec{p}_f = \vec{p}_i + \frac{\vec{F}}{m}t$ In code `p += F/m * dt` • Use $\vec{v} = \frac{\vec{p}}{m}$ (def'n of momentum) • Use $\Delta\vec{r} = \vec{v}\Delta t$ ($\vec{v} = \frac{\Delta\vec{r}}{\Delta t}$) Equivalent to $\vec{r}_f = \vec{r}_i + \vec{v}t$ In code `ball.pos += v*dt` • Increment time using `t += dt`

Most common screw-ups on tomorrow's test will probably be:

- using a constant or POI without defining it (or before defining it) in the code
- forgetting how to reference attributes (remember doing `ball.pos` or `ball.charge`???)
- forgetting to write vec(0,0,0) instead of just (0,0,0)
- forgetting you can add vector quantities together (i.e. `E_tot = E_1 + E_2`)
- forgetting you can get magnitude using `mag( )`
- to display a result you must with draw an arrow/text bow OR use a print statement (e.e. `print( )`)
- flipping the order of terms in the computation of $\vec{r}$
- screwing up the order of ECM (requires strict order to ensure conservation of energy)
- using `+=` and `=` inappropriately in ECM
- forgetting to increment time at end of ECM