Homework 8

Problem 1 (1 point)

Write a function to compute the volume of a sphere given it's radius. Get π from numpy.

In [8]:

```
from numpy import pi
def vol_sphere(radius):
    return 4/3 * radius**3 * pi      # Volume of a Sphere: 4pi/3 * r^3
```

Problem 2 (2 points)

Write a function to compute the vapor pressure of a species given A, B, and C according to the following equation

$$P_i^{sat} = \exp\left(A + \frac{B}{T+C}\right).$$
Assume T has units of K. Include a docstring documenting the function. Use numpy for $\exp$.

In [10]:

```
from numpy import exp
def pres_vapor(A, B, C, T):
    """
    Calculates the vapor pressure of a species given A, B, and C, with T for temperatur
    """
    return exp(A + B/(T+C))
```

Problem 3 (2 points)

A function can return more than two quantities, say `x1` and `x2` if you use the following return statement `return x1, x2` , where `x1` and `x2` are variables or expressions. You would the call the function as `x1,x2 = f(a,b,c)` .

Write a function that will return the two roots of a quadratic equation $ax^2+bx+c=0$, given a, b, and c. Assume the roots are real.

Try out the function for x^2-5x+5.

In [24]:

```
from numpy import sqrt
def quadratic_roots(a, b, c):
    """
    Returns the quadratic roots of an expression ax^2 + bx + c = 0, given a, b, and c.
    """
    x1 = (-b + sqrt(b**2 - 4*a*c))/(2*a)
    x2 = (-b - sqrt(b**2 - 4*a*c))/(2*a)
    return x1,x2
```

Problem 4 (3 points)

The first derivative of a function $f(x)$ can be approximated as $$\frac{df(x)}{dx} \approx \frac{f(x+\Delta x) - f(x)}{\Delta x}.$$ A good estimate for Δx is $$\Delta x = \epsilon|x|,$$ where ϵ is a small number.

Write a function called `derivative` that takes `f`, `x`, and `eps` as arguments and returns the derivative evaluated at `x`. Here, `f` is a function, `x` is where you are evaluating the derivative, and `eps` is ϵ in the above equation. Give `eps` a default value of 10^{-8}.

Test it out on the following

- $f(x) = x^2$ for $x=3$.
- $f(x) = \sin(x)$ for $x=\pi/2$
- $f(x) = \exp(x)$ for $x=1.01325$

In each case, compare to the exact solution.

For $\sin$ and $\exp$, you can use numpy and call the derivative function like so

```
d = derivative(np.sin, np.pi/2)
d = derivative(np.exp, 1.01325)
```

For $f(x)=x^2$, you will need to write your own function that you can pass to the `derivative` function.

In [39]:

```
from numpy import sin, exp, pi
def derivative(f, x, esp = 10**-8):
    delta_x = esp*abs(x)
    return (f(x + delta_x)-f(x))/delta_x

print("The derivative of x^2 for x = 3 is approximately", derivative(lambda x: x**2, 3)
print("The derivative of sin(x) for x = pi/2 is approximately", derivative(lambda x: si
print("The derivative of e^x for x = 1.01325 is approximately", derivative(lambda x: ex
```

```
The derivative of x^2 for x = 3 is approximately 6.000000022747068
The derivative of sin(x) for x = pi/2 is approximately -7.067899292141148e-09
The derivative of e^x for x = 1.01325 is approximately 2.7545387414656375
```