CP.1.1.1, Sauer3

Use the Bisection Method to find the root to six correct decimal places for
(a) $x^3 = 9$, (b) $3x^3 + x^2 = x + 5$, (c) $\cos^2(x) + 6 = x$.

CP.1.1.1, Sauer3, solution, Langou

- See
  https://colab.research.google.com/drive/10iYN2gfKaqoFF8Q4RDa3Y7569rp0Yrak
- Also, you can find a longer solution with some plots at
  https://colab.research.google.com/drive/1w9rst-uC52HqrB9TUBTFc9AGchog384R
- Our `bisect` Python code is at: http://math.ucdenver.edu/~langou/4650/4650.git/bucket/bisect.py.html

```
from math import cos
import scipy.optimize
import numpy as np
```

```
# (a) x^3 = 9

f = lambda x : ( x ** 3 ) - 9.

print( ( 9. )**(1/3) )

x_fsolve = scipy.optimize.fsolve( f, 1. )[0]
print( x_fsolve )

r = np.roots( [ 1., 0., 0., -9.] )
x_roots = np.real(r[np.isreal(r)])[0]
print( x_roots )

x_bisec, err_bound, numeval = bisect( f, 2., 3., 1e-6, False )
print( x_bisec, err_bound[-1], numeval )
```

```
2.080083823051904
2.080083823051904
2.0800838230519036
2.080082893371582 9.5367431640625e-07 21
```

```
# (b) 3x^3 + x^2 = x + 5

f = lambda x : 3. * ( x ** 3 ) + ( x ** 2 ) - x - 5.

print( 1./9. * ( - 1. + (593.-27.*(481.)**(1/2)) ** (1/3) \
                      + (593.+27.*(481.)**(1/2)) ** (1/3) ) )

x_fsolve = scipy.optimize.fsolve( f, 1. )[0]
print( x_fsolve )

r = np.roots( [ 3., 1., -1., -5.] )
x_roots = np.real(r[np.isreal(r)])[0]
print( x_roots )

x_bisec, err_bound, numeval = bisect( f, 1., 2., 1e-6, False )
print( x_bisec, err_bound[-1], numeval )
```

```
1.169726219853722
1.1697262198537242
1.1697262198537242
1.1697263717651367 9.5367431640625e-07 21
```

```
# (c) cos^2(x) + 6 = x

f = lambda x : ( cos(x) )**2 + 6. - x

import scipy.optimize
x_fsolve = scipy.optimize.fsolve( f, 1. )[0]
print( x_fsolve )

x_bisec, err_bound, numeval = bisect( f, 6., 7., 1e-6, False )
print( x_bisec, err_bound[-1], numeval )
```

```
6.776092316319578
6.776091575622559 9.5367431640625e-07 21
```