Homework 2 Solutions

1. accurately evaluating sin x - x when x is near 0.

a The relative round-off error is

$$E_{RO} \equiv \frac{\Big(\sin[x(1+\delta_1)](1+\delta_2) - x(1+\delta_1)\Big)(1+\delta_3) - (\sin x - x)}{(\sin x - x)}$$

Now, in order to approximate this we can use a Taylor polynomial for $\sin x$, expanding at 0:

$$\sin x \approx x - \frac{x^3}{6} \quad \text{for } |x| \text{ small.}$$

We could plough ahead, but to save work, observe that if any term linear in x survives in the numerator, it will dominate terms in x^3 or higher order. So it makes sense to try $\sin x \approx x$ in the numerator first, and include the x^3 term only if we end up with 0:

$$\text{numerator} \approx \Big(x(1+\delta_1)(1+\delta_2) - x(1+\delta_1)\Big)(1+\delta_3) - 0$$

$$= (\cancel{x} + \delta_1 x + \delta_2 x + \delta_1\delta_2 x - \cancel{x} - \delta_1 x)(1+\delta_3)$$

$$\approx \delta_2 x \qquad \text{because } |\delta_1| \ll 1 \text{ and } |\delta_3| \ll 1.$$

In the denominator, the linear terms cancel so we must include the cubic, using $\sin x - x \approx -x^3/6$.

Thus

$$E_{RO} \approx \frac{\delta_2 x}{-x^3/6} = \frac{-6\delta_2}{x^2}$$

and finally

$$|E_{RO}| \le \frac{6\,\frac{\varepsilon_{mach}}{2}}{x^2} = \boxed{\frac{3\varepsilon_{mach}}{x^2}}$$

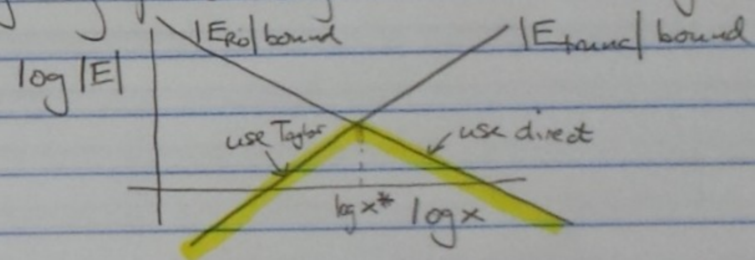
b Using Taylor's theorem, we have

$$f(x) \equiv \sin x - x = \frac{-x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{0x^8}{8!} + \cos\xi \cdot \frac{x^9}{9!}$$

where ξ is between 0 and x, because $f^{(9)}(x) = \cos x$.
Since $|\cos\xi| \le 1$ regardless of what ξ is, we have
$|$absolute truncation error$| \le |x|^9/9!$, and relative error

$$|E_{trunc}| \le \frac{|x^9|/9!}{|x|^3/3!} = \frac{6|x|^6}{9!}$$

c Since the bounds for $|E_{RO}|$ and $|E_{trunc}|$ are both powers of x, on a log-log plot they are both straight lines:



The cross-over point (where they are equally bad) is given by

$$\frac{3\varepsilon_{mach}}{|x^*|^2} = \frac{6|x^*|^6}{9!} \quad \text{or} \quad |x^*|^8 = \frac{9!\,\varepsilon_{mach}}{2}$$

or $$|x^*| = \left(\frac{9! \cdot 2^{-52}}{2}\right)^{\frac{1}{8}} \approx .05$$

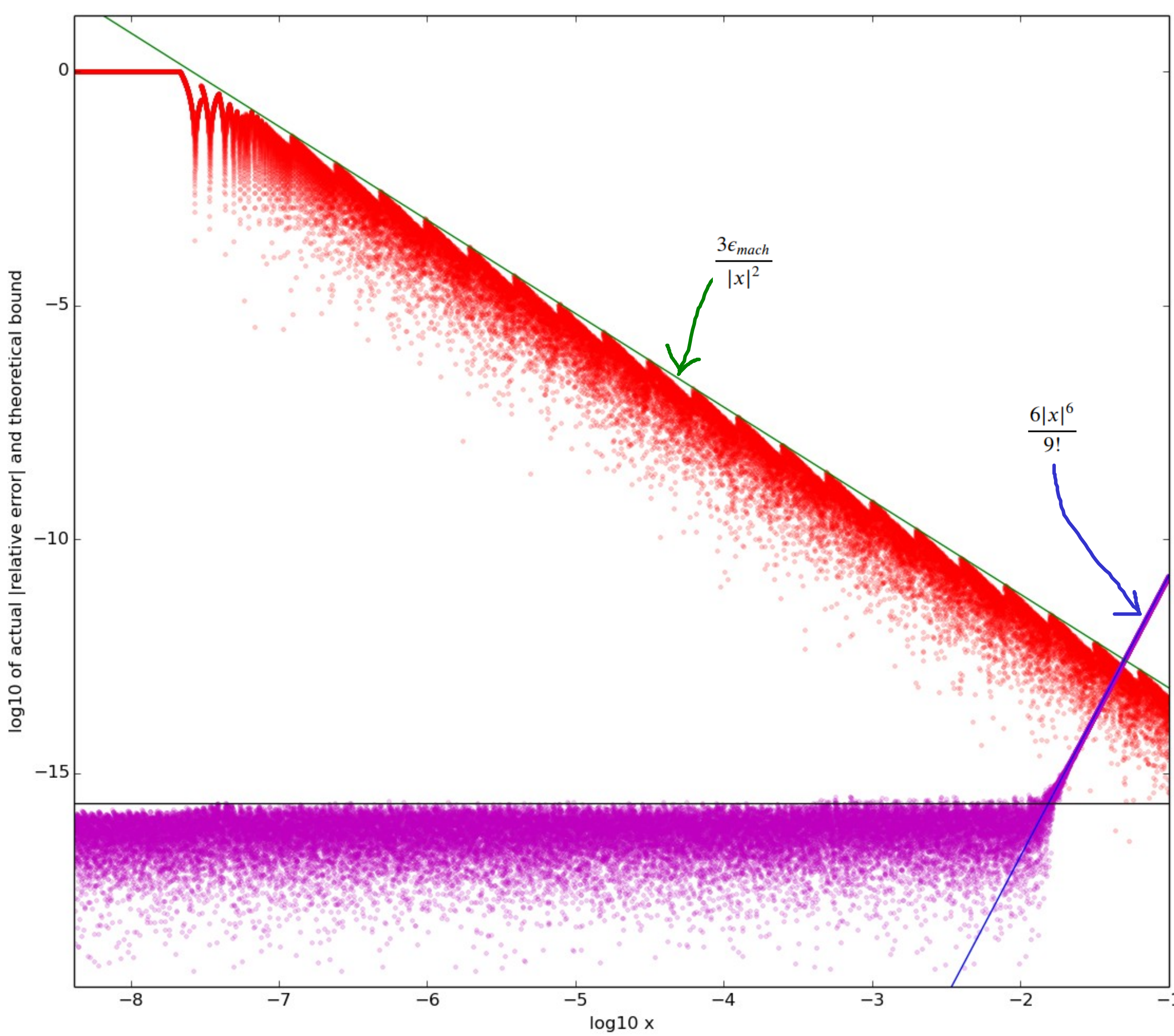
At the cross-over point, both relative error bounds are

$$|E_{worst}| = \frac{6|x^*|^6}{9!} \approx 2.6 \times 10^{-13}.$$

$\log_{10}|E_{worst}| \approx -12.5$, so we have about 12 reliable digits in the worst region.

Bonus information: empirical vaildation of the bounds derived above:

Red dots are errors with direct evaluation sin x - x.
Note that round-off is felt even with the Taylor polynomial (magenta dots) when the truncation error drops below machine epsilon (black horizontal line).



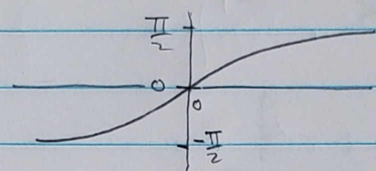
2. (a) Ackleh 2.10.1(a)

HW2 Ackleh 2.10.1(a)

2a

Bisection for root of arctan on $G = [-4.9, 5.1]$

- Yes, hypotheses are met:
  - $\arctan \in C([-4.9, 5.1])$ (in fact $\in C(\mathbb{R})$)
  - $\arctan(-4.9) < 0$, $\arctan(5.1) > 0$.



- If $x_k \equiv \frac{a_k + b_k}{2}$, then $|e_k| \equiv |x_k - z| \leq \frac{|a_k - b_k|}{2}$

i.e. $|e_k| \leq \frac{|a_0 - b_0|}{2 \cdot 2^k}$

For $|e_k| \leq 10^{-2}$, this is assured if

$$\frac{|a_0 - b_0|}{2 \cdot 2^k} \leq 10^{-2}$$

$$2^k \geq \frac{|-4.9 - 5.1|}{2 \cdot 10^{-2}} = \frac{10.0}{2 \cdot 10^{-2}} = 500$$

$$k \geq \log_2(500) = \frac{\log 500}{\log 2} \in [8.96, 8.97].$$

Thus $\boxed{\text{9 iterations suffice.}}$

(a) Program Algorithm 2.1 in your chosen programming language. Print a_k, b_k, $f(a_k)$, $f(b_k)$, and $f(x_k)$ for each step, so you can see what is happening.

(b) Try to solve $f(x) = 0$ with $\epsilon = 10^{-2}$, $\epsilon = 10^{-4}$, $\epsilon = 10^{-8}$, $\epsilon = 10^{-16}$, $\epsilon = 10^{-32}$, $\epsilon = 10^{-64}$, and $\epsilon = 10^{-128}$.

i. For each ϵ, compute the k at which the algorithm should stop.

ii. What behavior do you actually observe in the algorithm? Can you explain this behavior?

2. (b). Ackleh 2.10.3

```python
from numpy import sign,log,abs,nan

def bisection( f, a, b, tol ): # return approximate root and number of iterations take

    k_expected = int( log(abs(b-a)/tol/2)/log(2) ) + 1
    print(k_expected,'iterations expected')

    sa = sign(f(a))
    sb = sign(f(b))
    if sa==0: return a,0  # lucky direct hit
    if sb==0: return b,0  # lucky direct hit
    if sa==sb:
        print('Bad data given: signs of f at a and at b should be different')
        return nan,0
    k = 0
    print('\n  k uncertainty           a                    c                    b              f(a)      f(c)      f(b)')

    while abs(a-b)/2>tol:  # uncertainty > tol? do another iteration
        c = (a+b)/2
        if c==a or c==b:
            print('\nWARNING! a,b seem to be adjacent machine numbers: requested tolerance cannot be achieved with machine numbers')
            return c,k
        fc = f(c)
        sc = sign(f(c))
        print(f"{k:3d}  {abs(a-b)/2:.2e}  {a:.15e} {c:.15e} {b:.15e} "+
              f"{f(a):+.2e} {fc:+.2e} {f(b):+.2e}")
        if fc==0: return c,k  # lucky direct hit
        if sa == sc:
            a = c
        else:
            b = c
        k += 1
    c = (a+b)/2
    print(f"{k:3d}  {abs(a-b)/2:.2e}  {a:.15e} {c:.15e} {b:.15e} "+
          f"{f(a):+.2e} {f(c):+.2e} {f(b):+.2e}")
    return c,k
```

```python
# Ackleh 2.10.3

def f(x): return x**2 - 2
a,b = 1,2
for eps in [10**(-2**i) for i in range(1,5)]:
    print('\n\nepsilon =',eps)
    x,k = bisection(f,a,b,eps)
    print(x,'is approximate root')
    print(k,'iterations used')
```

```
epsilon = 0.01
6 iterations expected

  k uncertainty           a                    c                    b              f(a)      f(c)      f(b)
  0  5.00e-01  1.000000000000000e+00 1.500000000000000e+00 2.000000000000000e+00 -1.00e+00 +2.50e-01 +2.00e+00
  1  2.50e-01  1.000000000000000e+00 1.250000000000000e+00 1.500000000000000e+00 -1.00e+00 -4.38e-01 +2.50e-01
  2  1.25e-01  1.250000000000000e+00 1.375000000000000e+00 1.500000000000000e+00 -4.38e-01 -1.09e-01 +2.50e-01
  3  6.25e-02  1.375000000000000e+00 1.437500000000000e+00 1.500000000000000e+00 -1.09e-01 +6.64e-02 +2.50e-01
  4  3.12e-02  1.375000000000000e+00 1.406250000000000e+00 1.437500000000000e+00 -1.09e-01 -2.25e-02 +6.64e-02
  5  1.56e-02  1.406250000000000e+00 1.421875000000000e+00 1.437500000000000e+00 -2.25e-02 +2.17e-02 +6.64e-02
  6  7.81e-03  1.406250000000000e+00 1.414062500000000e+00 1.421875000000000e+00 -2.25e-02 -4.27e-04 +2.17e-02
1.4140625 is approximate root
6 iterations used


epsilon = 0.0001
13 iterations expected

  k uncertainty           a                    c                    b              f(a)      f(c)      f(b)
  0  5.00e-01  1.000000000000000e+00 1.500000000000000e+00 2.000000000000000e+00 -1.00e+00 +2.50e-01 +2.00e+00
  1  2.50e-01  1.000000000000000e+00 1.250000000000000e+00 1.500000000000000e+00 -1.00e+00 -4.38e-01 +2.50e-01
  2  1.25e-01  1.250000000000000e+00 1.375000000000000e+00 1.500000000000000e+00 -4.38e-01 -1.09e-01 +2.50e-01
  3  6.25e-02  1.375000000000000e+00 1.437500000000000e+00 1.500000000000000e+00 -1.09e-01 +6.64e-02 +2.50e-01
  4  3.12e-02  1.375000000000000e+00 1.406250000000000e+00 1.437500000000000e+00 -1.09e-01 -2.25e-02 +6.64e-02
  5  1.56e-02  1.406250000000000e+00 1.421875000000000e+00 1.437500000000000e+00 -2.25e-02 +2.17e-02 +6.64e-02
  6  7.81e-03  1.406250000000000e+00 1.414062500000000e+00 1.421875000000000e+00 -2.25e-02 -4.27e-04 +2.17e-02
  7  3.91e-03  1.414062500000000e+00 1.417968750000000e+00 1.421875000000000e+00 -4.27e-04 +1.06e-02 +2.17e-02
  8  1.95e-03  1.414062500000000e+00 1.416015625000000e+00 1.417968750000000e+00 -4.27e-04 +5.10e-03 +1.06e-02
  9  9.77e-04  1.414062500000000e+00 1.415039062500000e+00 1.416015625000000e+00 -4.27e-04 +2.34e-03 +5.10e-03
 10  4.88e-04  1.414062500000000e+00 1.414550781250000e+00 1.415039062500000e+00 -4.27e-04 +9.54e-04 +2.34e-03
 11  2.44e-04  1.414062500000000e+00 1.414306640625000e+00 1.414550781250000e+00 -4.27e-04 +2.63e-04 +9.54e-04
 12  1.22e-04  1.414062500000000e+00 1.414184570312500e+00 1.414306640625000e+00 -4.27e-04 -8.20e-05 +2.63e-04
 13  6.10e-05  1.414184570312500e+00 1.414245605468750e+00 1.414306640625000e+00 -8.20e-05 +9.06e-05 +2.63e-04
1.41424560546875 is approximate root
13 iterations used
```

```
epsilon = 1e-08
26 iterations expected

  k uncertainty           a                     c                     b                  f(a)      f(c)      f(b)
  0  5.00e-01  1.000000000000000e+00 1.500000000000000e+00 2.000000000000000e+00 -1.00e+00 +2.50e-01 +2.00e+00
  1  2.50e-01  1.000000000000000e+00 1.250000000000000e+00 1.500000000000000e+00 -1.00e+00 -4.38e-01 +2.50e-01
  2  1.25e-01  1.250000000000000e+00 1.375000000000000e+00 1.500000000000000e+00 -4.38e-01 -1.09e-01 +2.50e-01
  3  6.25e-02  1.375000000000000e+00 1.437500000000000e+00 1.500000000000000e+00 -1.09e-01 +6.64e-02 +2.50e-01
  4  3.12e-02  1.375000000000000e+00 1.406250000000000e+00 1.437500000000000e+00 -1.09e-01 -2.25e-02 +6.64e-02
  5  1.56e-02  1.406250000000000e+00 1.421875000000000e+00 1.437500000000000e+00 -2.25e-02 +2.17e-02 +6.64e-02
  6  7.81e-03  1.406250000000000e+00 1.414062500000000e+00 1.421875000000000e+00 -2.25e-02 -4.27e-04 +2.17e-02
  7  3.91e-03  1.414062500000000e+00 1.417968750000000e+00 1.421875000000000e+00 -4.27e-04 +1.06e-02 +2.17e-02
  8  1.95e-03  1.414062500000000e+00 1.416015625000000e+00 1.417968750000000e+00 -4.27e-04 +5.10e-03 +1.06e-02
  9  9.77e-04  1.414062500000000e+00 1.415039062500000e+00 1.416015625000000e+00 -4.27e-04 +2.34e-03 +5.10e-03
 10  4.88e-04  1.414062500000000e+00 1.414550781250000e+00 1.415039062500000e+00 -4.27e-04 +9.54e-04 +2.34e-03
 11  2.44e-04  1.414062500000000e+00 1.414306640625000e+00 1.414550781250000e+00 -4.27e-04 +2.63e-04 +9.54e-04
 12  1.22e-04  1.414062500000000e+00 1.414184570312500e+00 1.414306640625000e+00 -4.27e-04 -8.20e-05 +2.63e-04
 13  6.10e-05  1.414184570312500e+00 1.414245605468750e+00 1.414306640625000e+00 -8.20e-05 +9.06e-05 +2.63e-04
 14  3.05e-05  1.414184570312500e+00 1.414215087890625e+00 1.414245605468750e+00 -8.20e-05 +4.31e-06 +9.06e-05
 15  1.53e-05  1.414184570312500e+00 1.414199829101562e+00 1.414215087890625e+00 -8.20e-05 -3.88e-05 +4.31e-06
 16  7.63e-06  1.414199829101562e+00 1.414207458496094e+00 1.414215087890625e+00 -3.88e-05 -1.73e-05 +4.31e-06
 17  3.81e-06  1.414207458496094e+00 1.414211273193359e+00 1.414215087890625e+00 -1.73e-05 -6.47e-06 +4.31e-06
 18  1.91e-06  1.414211273193359e+00 1.414213180541992e+00 1.414215087890625e+00 -6.47e-06 -1.08e-06 +4.31e-06
 19  9.54e-07  1.414213180541992e+00 1.414214134216309e+00 1.414215087890625e+00 -1.08e-06 +1.62e-06 +4.31e-06
 20  4.77e-07  1.414213180541992e+00 1.414213657379150e+00 1.414214134216309e+00 -1.08e-06 +2.69e-07 +1.62e-06
 21  2.38e-07  1.414213180541992e+00 1.414213418960571e+00 1.414213657379150e+00 -1.08e-06 -4.06e-07 +2.69e-07
 22  1.19e-07  1.414213418960571e+00 1.414213538169861e+00 1.414213657379150e+00 -4.06e-07 -6.85e-08 +2.69e-07
 23  5.96e-08  1.414213538169861e+00 1.414213597774506e+00 1.414213657379150e+00 -6.85e-08 +1.00e-07 +2.69e-07
 24  2.98e-08  1.414213538169861e+00 1.414213567972183e+00 1.414213597774506e+00 -6.85e-08 +1.58e-08 +1.00e-07
 25  1.49e-08  1.414213538169861e+00 1.414213553071022e+00 1.414213567972183e+00 -6.85e-08 -2.63e-08 +1.58e-08
 26  7.45e-09  1.414213553071022e+00 1.414213560521603e+00 1.414213567972183e+00 -2.63e-08 -5.24e-09 +1.58e-08
1.4142135605216026 is approximate root
26 iterations used
```

```
epsilon = 1e-16
53 iterations expected

  k uncertainty           a                     c                     b                  f(a)      f(c)      f(b)
  0  5.00e-01  1.000000000000000e+00 1.500000000000000e+00 2.000000000000000e+00 -1.00e+00 +2.50e-01 +2.00e+00
  1  2.50e-01  1.000000000000000e+00 1.250000000000000e+00 1.500000000000000e+00 -1.00e+00 -4.38e-01 +2.50e-01
  2  1.25e-01  1.250000000000000e+00 1.375000000000000e+00 1.500000000000000e+00 -4.38e-01 -1.09e-01 +2.50e-01
  3  6.25e-02  1.375000000000000e+00 1.437500000000000e+00 1.500000000000000e+00 -1.09e-01 +6.64e-02 +2.50e-01
  4  3.12e-02  1.375000000000000e+00 1.406250000000000e+00 1.437500000000000e+00 -1.09e-01 -2.25e-02 +6.64e-02
  5  1.56e-02  1.406250000000000e+00 1.421875000000000e+00 1.437500000000000e+00 -2.25e-02 +2.17e-02 +6.64e-02
  6  7.81e-03  1.406250000000000e+00 1.414062500000000e+00 1.421875000000000e+00 -2.25e-02 -4.27e-04 +2.17e-02
  7  3.91e-03  1.414062500000000e+00 1.417968750000000e+00 1.421875000000000e+00 -4.27e-04 +1.06e-02 +2.17e-02
  8  1.95e-03  1.414062500000000e+00 1.416015625000000e+00 1.417968750000000e+00 -4.27e-04 +5.10e-03 +1.06e-02
  9  9.77e-04  1.414062500000000e+00 1.415039062500000e+00 1.416015625000000e+00 -4.27e-04 +2.34e-03 +5.10e-03
 10  4.88e-04  1.414062500000000e+00 1.414550781250000e+00 1.415039062500000e+00 -4.27e-04 +9.54e-04 +2.34e-03
 11  2.44e-04  1.414062500000000e+00 1.414306640625000e+00 1.414550781250000e+00 -4.27e-04 +2.63e-04 +9.54e-04
 12  1.22e-04  1.414062500000000e+00 1.414184570312500e+00 1.414306640625000e+00 -4.27e-04 -8.20e-05 +2.63e-04
 13  6.10e-05  1.414184570312500e+00 1.414245605468750e+00 1.414306640625000e+00 -8.20e-05 +9.06e-05 +2.63e-04
 14  3.05e-05  1.414184570312500e+00 1.414215087890625e+00 1.414245605468750e+00 -8.20e-05 +4.31e-06 +9.06e-05
 15  1.53e-05  1.414184570312500e+00 1.414199829101562e+00 1.414215087890625e+00 -8.20e-05 -3.88e-05 +4.31e-06
 16  7.63e-06  1.414199829101562e+00 1.414207458496094e+00 1.414215087890625e+00 -3.88e-05 -1.73e-05 +4.31e-06
 17  3.81e-06  1.414207458496094e+00 1.414211273193359e+00 1.414215087890625e+00 -1.73e-05 -6.47e-06 +4.31e-06
 18  1.91e-06  1.414211273193359e+00 1.414213180541992e+00 1.414215087890625e+00 -6.47e-06 -1.08e-06 +4.31e-06
 19  9.54e-07  1.414213180541992e+00 1.414214134216309e+00 1.414215087890625e+00 -1.08e-06 +1.62e-06 +4.31e-06
 20  4.77e-07  1.414213180541992e+00 1.414213657379150e+00 1.414214134216309e+00 -1.08e-06 +2.69e-07 +1.62e-06
 21  2.38e-07  1.414213180541992e+00 1.414213418960571e+00 1.414213657379150e+00 -1.08e-06 -4.06e-07 +2.69e-07
 22  1.19e-07  1.414213418960571e+00 1.414213538169861e+00 1.414213657379150e+00 -4.06e-07 -6.85e-08 +2.69e-07
 23  5.96e-08  1.414213538169861e+00 1.414213597774506e+00 1.414213657379150e+00 -6.85e-08 +1.00e-07 +2.69e-07
 24  2.98e-08  1.414213538169861e+00 1.414213567972183e+00 1.414213597774506e+00 -6.85e-08 +1.58e-08 +1.00e-07
 25  1.49e-08  1.414213538169861e+00 1.414213553071022e+00 1.414213567972183e+00 -6.85e-08 -2.63e-08 +1.58e-08
 26  7.45e-09  1.414213553071022e+00 1.414213560521603e+00 1.414213567972183e+00 -2.63e-08 -5.24e-09 +1.58e-08
 27  3.73e-09  1.414213560521603e+00 1.414213564246893e+00 1.414213567972183e+00 -5.24e-09 +5.30e-09 +1.58e-08
 28  1.86e-09  1.414213560521603e+00 1.414213562384248e+00 1.414213564246893e+00 -5.24e-09 +3.15e-11 +5.30e-09
 29  9.31e-10  1.414213560521603e+00 1.414213561452925e+00 1.414213562384248e+00 -5.24e-09 -2.60e-09 +3.15e-11
 30  4.66e-10  1.414213561452925e+00 1.414213561918586e+00 1.414213562384248e+00 -2.60e-09 -1.29e-09 +3.15e-11
 31  2.33e-10  1.414213561918586e+00 1.414213562151417e+00 1.414213562384248e+00 -1.29e-09 -6.27e-10 +3.15e-11
 32  1.16e-10  1.414213562151417e+00 1.414213562267832e+00 1.414213562384248e+00 -6.27e-10 -2.98e-10 +3.15e-11
 33  5.82e-11  1.414213562267832e+00 1.414213562326040e+00 1.414213562384248e+00 -2.98e-10 -1.33e-10 +3.15e-11
 34  2.91e-11  1.414213562326040e+00 1.414213562355144e+00 1.414213562384248e+00 -1.33e-10 -5.08e-11 +3.15e-11
 35  1.46e-11  1.414213562355144e+00 1.414213562369696e+00 1.414213562384248e+00 -5.08e-11 -9.61e-12 +3.15e-11
 36  7.28e-12  1.414213562369696e+00 1.414213562376972e+00 1.414213562384248e+00 -9.61e-12 +1.10e-11 +3.15e-11
 37  3.64e-12  1.414213562369696e+00 1.414213562373334e+00 1.414213562376972e+00 -9.61e-12 +6.75e-13 +1.10e-11
 38  1.82e-12  1.414213562369696e+00 1.414213562371515e+00 1.414213562373334e+00 -9.61e-12 -4.47e-12 +6.75e-13
 39  9.09e-13  1.414213562371515e+00 1.414213562372424e+00 1.414213562373334e+00 -4.47e-12 -1.90e-12 +6.75e-13
 40  4.55e-13  1.414213562372424e+00 1.414213562372879e+00 1.414213562373334e+00 -1.90e-12 -6.11e-13 +6.75e-13
 41  2.27e-13  1.414213562372879e+00 1.414213562373106e+00 1.414213562373334e+00 -6.11e-13 +3.24e-14 +6.75e-13
 42  1.14e-13  1.414213562372879e+00 1.414213562372993e+00 1.414213562373106e+00 -6.11e-13 -2.89e-13 +3.24e-14
 43  5.68e-14  1.414213562372993e+00 1.414213562373050e+00 1.414213562373106e+00 -2.89e-13 -1.29e-13 +3.24e-14
 44  2.84e-14  1.414213562373050e+00 1.414213562373078e+00 1.414213562373106e+00 -1.29e-13 -4.82e-14 +3.24e-14
 45  1.42e-14  1.414213562373078e+00 1.414213562373092e+00 1.414213562373106e+00 -4.82e-14 -7.99e-15 +3.24e-14
 46  7.11e-15  1.414213562373092e+00 1.414213562373099e+00 1.414213562373106e+00 -7.99e-15 +1.20e-14 +3.24e-14
 47  3.55e-15  1.414213562373092e+00 1.414213562373096e+00 1.414213562373099e+00 -7.99e-15 +2.22e-15 +1.20e-14
 48  1.78e-15  1.414213562373092e+00 1.414213562373094e+00 1.414213562373096e+00 -7.99e-15 -2.89e-15 +2.22e-15
 49  8.88e-16  1.414213562373094e+00 1.414213562373095e+00 1.414213562373096e+00 -2.89e-15 -4.44e-16 +2.22e-15
 50  4.44e-16  1.414213562373095e+00 1.414213562373095e+00 1.414213562373096e+00 -4.44e-16 +8.88e-16 +2.22e-15
 51  2.22e-16  1.414213562373095e+00 1.414213562373095e+00 1.414213562373095e+00 -4.44e-16 +4.44e-16 +8.88e-16

WARNING! a,b seem to be adjacent machine numbers: requested tolerance cannot be achieved with machine numbers
1.414213562373095 is approximate root
52 iterations used
```

There is a module ieee754 that allows us to print the machine codes in hex form.
This confirms our conclusion that a and b become adjacent machine numbers in the above:

Using ieee754 module to inspect machine codes

```python
from numpy import sign,log,abs,nan
from ieee754 import IEEE754  # library to do float to machine code conversion

def bisection( f, a, b, tol ): # return approximate root and number of iterations take

    k_expected = int( log(abs(b-a)/tol/2)/log(2) ) + 1
    print(k_expected,'iterations expected')

    sa = sign(f(a))
    sb = sign(f(b))
    if sa==0: return a,0  # lucky direct hit
    if sb==0: return b,0  # lucky direct hit
    if sa==sb:
        print('Bad data given: signs of f at a and at b should be different')
        return nan,0
    k = 0
    while abs(a-b)/2>tol:
        c = (a+b)/2
        print('\ta', IEEE754(a).str2hex() )
        print('\tc', IEEE754(c).str2hex() )
        print('\tb', IEEE754(b).str2hex() )
        if c==a or c==b:
            print('a,b seem to be adjacent machine numbers: requested tolerance cannot be achieved with machine numbers')
            return c,k
        fc = f(c)
        sc = sign(f(c))
        #print("{:5d} {:+.15f} {:+.15f} {:+.15f} {:+.15f} {:+.15f} {:2d}".format(k,a,b,f(a),f(b),f(c),int(sc)))
        print("{:5d} {:+.15e} {:+.15e} {:+.15e} {:+.15e} {:+.15e} ".format(k,a,b,f(a),f(b),f(c)))
        k += 1
        if fc==0: return c,k
        if sa == sc:
            a = c
        else:
            b = c
    return (a+b)/2,k
```

```python
# Ackleh 2.10.3
def f(x): return x**2 - 2
a,b = 1,2
for eps in [1e-16]:
    print('\nepsilon =',eps)
    x,k = bisection(f,a,b,eps)
    print(x,'is approximate root')
    print(k,'iterations used')
```

```
   48 +1.414213562373092e+00 +1.414213562373096e+00 -7.993605777301127e-15 +2.220446049250313e-15 -2.886579864025407e-15
        a 3ff6a09e667f3bc8
        c 3ff6a09e667f3bcc
        b 3ff6a09e667f3bd0
   49 +1.414213562373094e+00 +1.414213562373096e+00 -2.886579864025407e-15 +2.220446049250313e-15 -4.440892098500626e-16
        a 3ff6a09e667f3bcc
        c 3ff6a09e667f3bce
        b 3ff6a09e667f3bd0
   50 +1.414213562373095e+00 +1.414213562373096e+00 -4.440892098500626e-16 +2.220446049250313e-15 +8.881784197001252e-16
        a 3ff6a09e667f3bcc
        c 3ff6a09e667f3bcd
        b 3ff6a09e667f3bce
   51 +1.414213562373095e+00 +1.414213562373095e+00 -4.440892098500626e-16 +8.881784197001252e-16 +4.440892098500626e-16
        a 3ff6a09e667f3bcc
        c 3ff6a09e667f3bcc
        b 3ff6a09e667f3bcd
a,b seem to be adjacent machine numbers: requested tolerance cannot be achieved with machine numbers
1.414213562373095 is approximate root
52 iterations used
```

c. Ackleh 2.10.4   4th root of 1000

```python
# Ackleh 2.10.4
def f(x): return x**4 - 1000

a,b = 0,100
eps = 1e-5
print('\nepsilon =',eps)
x,k = bisection(f,a,b,eps)
print(x,'is approximate root')
print(k,'iterations used')
```

```
epsilon = 1e-05
23 iterations expected

  k uncertainty           a                     c                     b                f(a)      f(c)      f(b)
  0  5.00e+01  0.000000000000000e+00 5.000000000000000e+01 1.000000000000000e+02 -1.00e+03 +6.25e+06 +1.00e+08
  1  2.50e+01  0.000000000000000e+00 2.500000000000000e+01 5.000000000000000e+01 -1.00e+03 +3.90e+05 +6.25e+06
  2  1.25e+01  0.000000000000000e+00 1.250000000000000e+01 2.500000000000000e+01 -1.00e+03 +2.34e+04 +3.90e+05
  3  6.25e+00  0.000000000000000e+00 6.250000000000000e+00 1.250000000000000e+01 -1.00e+03 +5.26e+02 +2.34e+04
  4  3.12e+00  0.000000000000000e+00 3.125000000000000e+00 6.250000000000000e+00 -1.00e+03 -9.05e+02 +5.26e+02
  5  1.56e+00  3.125000000000000e+00 4.687500000000000e+00 6.250000000000000e+00 -9.05e+02 -5.17e+02 +5.26e+02
  6  7.81e-01  4.687500000000000e+00 5.468750000000000e+00 6.250000000000000e+00 -5.17e+02 -1.06e+02 +5.26e+02
  7  3.91e-01  5.468750000000000e+00 5.859375000000000e+00 6.250000000000000e+00 -1.06e+02 +1.79e+02 +5.26e+02
  8  1.95e-01  5.468750000000000e+00 5.664062500000000e+00 5.859375000000000e+00 -1.06e+02 +2.92e+01 +1.79e+02
  9  9.77e-02  5.468750000000000e+00 5.566406250000000e+00 5.664062500000000e+00 -1.06e+02 -3.99e+01 +2.92e+01
 10  4.88e-02  5.566406250000000e+00 5.615234375000000e+00 5.664062500000000e+00 -3.99e+01 -5.81e+00 +2.92e+01
 11  2.44e-02  5.615234375000000e+00 5.639648437500000e+00 5.664062500000000e+00 -5.81e+00 +1.16e+01 +2.92e+01
 12  1.22e-02  5.615234375000000e+00 5.627441406250000e+00 5.639648437500000e+00 -5.81e+00 +2.87e+00 +1.16e+01
 13  6.10e-03  5.615234375000000e+00 5.621337890625000e+00 5.627441406250000e+00 -5.81e+00 -1.48e+00 +2.87e+00
 14  3.05e-03  5.621337890625000e+00 5.624389648437500e+00 5.627441406250000e+00 -1.48e+00 +6.95e-01 +2.87e+00
 15  1.53e-03  5.621337890625000e+00 5.622863769531250e+00 5.624389648437500e+00 -1.48e+00 -3.91e-01 +6.95e-01
 16  7.63e-04  5.622863769531250e+00 5.623626708984375e+00 5.624389648437500e+00 -3.91e-01 +1.52e-01 +6.95e-01
 17  3.81e-04  5.622863769531250e+00 5.623245239257812e+00 5.623626708984375e+00 -3.91e-01 -1.20e-01 +1.52e-01
 18  1.91e-04  5.623245239257812e+00 5.623435974121094e+00 5.623626708984375e+00 -1.20e-01 +1.62e-02 +1.52e-01
 19  9.54e-05  5.623245239257812e+00 5.623340606689453e+00 5.623435974121094e+00 -1.20e-01 -5.17e-02 +1.62e-02
 20  4.77e-05  5.623340606689453e+00 5.623388290405273e+00 5.623435974121094e+00 -5.17e-02 -1.78e-02 +1.62e-02
 21  2.38e-05  5.623388290405273e+00 5.623412132263184e+00 5.623435974121094e+00 -1.78e-02 -7.96e-04 +1.62e-02
 22  1.19e-05  5.623412132263184e+00 5.623424053192139e+00 5.623435974121094e+00 -7.96e-04 +7.68e-03 +1.62e-02
 23  5.96e-06  5.623412132263184e+00 5.623418092727661e+00 5.623424053192139e+00 -7.96e-04 +3.44e-03 +7.68e-03
5.623418092727661 is approximate root
23 iterations used
```