1 Practice

Solution 1.1. When x is close to zero, the dominant term is $\ln(x)$ which is very negative and when x is large, the dominant term is x^{23} which is very positive. Therefore, we expect this function to have exactly one zero, and we'll choose $(a_0, b_0) = (0.01, 1)$ as the starting interval. (There are many other reasonable choices as well.)

Now computing for three rounds:

$$f(\frac{0.01+1}{2}) = 0.9390425378713009 > 0 \to (a_1, b_1) = (0.01, 0.505)$$

$$f(\frac{.01+.505}{2}) = -0.06670774912805477 \to (a_2, b_2) = (.2575, .505)$$

$$f(\frac{.2575+.505}{2}) = 0.48645055118309144 \to (a_3, b_3) = (.2575, .38125)$$

Solution 1.2. Computing three iterations with $x_0 = 0.3$ yields

$$x_1 = 0.1601475605933413$$

$$x_2 = 0.8189252978385426$$

$$x_3 = -1.0673553863575176$$

We cannot perform a fourth iteration because the domain of $f(x)$ does not include negative numbers.

Solution 1.3. Newton's method requires that we compute the derivative

$$f'(x) = 23x^{22} + \cos(x)e^{\sin(x)} + \frac{1}{x}$$

and then use the iteration function

$$h(x) = x - \frac{f(x)}{f'(x)}.$$

With these iterations, we obtain

$$x_1 = 0.26971014933637455$$

$$x_2 = 0.27073362325351646$$

$$x_3 = 0.27073498191247675$$

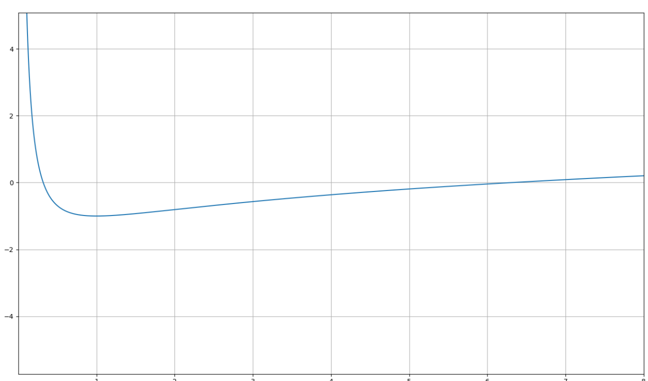
Solution 1.4. For the naïve fixed point function $g(x) = x - f(x)$, you can check that essentially no initial guesses lead to reasonable convergence. For Newton's method on the other hand, we obtain reasonably rapid convergence in few iterations. The difference is due to the good selection of an iteration function as well as starting point.

2 Bisection

Solution 2.1. Here is a sample bisection function. Note that we are using $b-a$ as an upper bound for the error $|a-x|$, where x is the true zero of the function that we are investigating.

```
def bisect(func, a, b, err):
 while b-a > err:
  midpoint = (a+b)/2
  if func(midpoint)*func(a) < 0:
   b = midpoint
  elif func(midpoint)*func(b) < 0:
   a = midpoint
  else:
   return (midpoint, 0)
 return (a, b-a)
```

Solution 2.2. In our bisection algorith, we start with a choice of a and b. If we want to use it to find all the zeroes, then we first need to study the function a bit in order to determine how many zeroes it has and where its zeroes are so that we can begin with reasonable initial guesses. In general, we won't be able to study a graph of our function to determine roughly where its zeroes are, but for this particular example we'll investigate a graph to figure out where to run bisection.



From a graph, we can see that there is a zero in the interval $[0.1, 1]$ and a zero in the interval $[6, 7]$. Thus, we can run bisection with f on the interval $[0.1, 1]$ and the interval $[6, 7]$.

```
from math import log
def f(x):
 return 1/x+log(x)-2
```

Then for the first zero, using the error tolerance of 2^{-10}, we obtain

$$(\text{approx, error}) = (0.31708984375000004, 0.0008789062499999778).$$

For the second zero, using the error tolerance of 2^{-10}, we obtain

$$(\text{approx, error}) = (6.3046875, 0.0007812500000001776).$$

We could of course obtain a better approximation by asking for a smaller error, but be careful in asking for too small an error. The best result we can obtain here is limited by floating point roundoff error, and if we ask for too small an error, bisection will either run forever (because it cannot reach the necessary error tolerance) or it will erroneously report an error of 0.

3 Approximating π

Solution 3.1. Using bisection on $f(x) = sin(x)$ with $a = 3, b = 4$ and an error tolerance of 2^{-51} obtains

$$(3.141592653589793, 4.440892098500626e - 16, 51).$$

Note that we *cannot* ask for a smaller error here otherwise bisection will not terminate.

Solution 3.2. According to Newton's method, we should take $g(x) = x - \frac{f(x)}{f'(x)}$ as our fixed point iteration. In this case, $f(x) = \sin(x)$ and $f'(x) = \cos(x)$, so we get $g(x) = x - \tan(x)$. Iterating this three times starting from the initial guess $x_0 = 3$ yields

$$x_3 = 3.141592653589793$$

which is comparable with $fl(\pi)$. It turns out that the iteration $g(x) = x + \sin(x)$ is also very effective, due to the fact that $\cos(\pi) = -1$. Fixed point iteration was very effective here because we were able to obtain quadratic convergence of our iterations (rather than linear convergence as in bisection) due to a good choice of iteration function and initial value.

Solution 3.3. The most expensive steps in bisection and fixed point iteration are the evalution of the function. In particular in approximating π, obtaining $\sin(x)$ (and $\cos(x)$ or $\tan(x)$) to the accuracy required for this application requires quite a few operations (for example using the Taylor series approximations). This is by far the dominant contribution to the computational time and would not be helpful for computing a large number of digits of π, since the accuracy at each step doesn't scale linearly, so we may need many more terms to obtain a small increase in accuracy.

Solution 3.4. To begin studying the relative errors, we first investigate the exact error $y_{n+1} - x$.

$$\begin{aligned} y_{n+1} - x &= fl(g(y_n)) - g(x) \\ &= (fl(g(y_n)) - g(y_n)) + (g(y_n) - g(x)) \end{aligned}$$

The first term is a roundoff error and the second term can be handled using the MVT.

$$\begin{aligned} &\leq |g(y_n)|\varepsilon + g'(\xi_n)(y_n - x) \\ &= |g(y_n) - x + x|\varepsilon + g'(\xi_n)(y_n - x) \\ &< |g(y_n) - g(x)|\varepsilon + |x|\varepsilon + \frac{y_n - x}{2} \end{aligned}$$

Applying the mean value theorem one more time gives

$$< |y_n - x|\frac{1+\varepsilon}{2} + |x|\varepsilon$$

Finally, dividing through the whole inequality by $|x|$, we obtain the inequality

$$r_{n+1} \leq r_n\frac{1+\varepsilon}{2} + \varepsilon.$$

Solution 3.5. Applying the inequality successively, we obtain

$$\begin{aligned}
r_{n+1} &\leq \theta r_n + \varepsilon \\
&\leq \theta(\theta r_{n-1} + \varepsilon) + \varepsilon \\
&= \theta^2 r_{n-1} + \varepsilon(1+\theta) \\
&\leq \theta^2(\theta r_{n-2} + \varepsilon) + \varepsilon(1+\theta) \\
&= \theta^3 r_{n-2} + \varepsilon(1+\theta+\theta^2) \\
&\vdots \\
&\leq \theta^{n+1} r_0 + \varepsilon(1+\theta+\cdots+\theta^n) \\
&\leq \theta^{n+1} r_0 + \varepsilon(1+\theta+\theta^2+\dots) \\
&= \theta^{n+1} r_0 + \frac{\varepsilon}{1-\theta}.
\end{aligned}$$