AMSC460: Homework 1

JAMES ZHANG*

September 9, 2024

3. (a) How many distinct positive numbers can be represented in a floating point system using base $\beta = 10$, precision $t = 2$ and exponent range $L = -9$, $U = 10$?
(Assume normalized fractions and don't worry about underflow.)

Solution. In general, a floating point number can be expressed in the representation

$$fl(x) = \pm \left(\frac{\widetilde{d}_0}{\beta^0} + \frac{\widetilde{d}_1}{\beta^1} + \cdots + \frac{\widetilde{d}_{t-1}}{\beta^{t-1}} \right) \times \beta^e$$

The problem statement specifies that we are looking for positive integers, $t = 2$, $\beta = 10$, and e is bounded by -9 and 10. Applying this information, we now have the more specific representation

$$fl(x) = + \left(\widetilde{d}_0 + \frac{\widetilde{d}_1}{10} \right) \times 10^e$$

Since we assume normalized fractions, $\widetilde{d}_0 \neq 0$, so it can attain the digits $1 - 9$, or 9 possibilties. $\widetilde{d}_1$ can be any digit, so 10 possibilities. Finally, e can be any number from -9 to 10, so 20 possibilities. Multiplying these together yields

$$9 \times 10 \times 20 = 1800 \text{ distinct positive integers}$$

□

*Email: jzhang72@terpmail.umd.edu

13. Consider the linear system

$$\begin{pmatrix} a & b \\ b & a \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

with $a, b > 0$; $a \neq b$.

(a) If $a \approx b$, what is the numerical difficulty in solving this linear system?

Solution. To solve this system, if the square matrix is invertble, we would invert the matrix and solve for the vector $\begin{pmatrix} x & y \end{pmatrix}^T$. By the Invertible Matrix Theorem, one of the conditioning for checking if a matrix is invertible is if its determinant is nonzero. Note that the determinant of the square matrix is

$$\det \begin{pmatrix} a & b \\ b & a \end{pmatrix} = a^2 - b^2 = (a-b)(a+b)$$

If $a \approx b$, and specifically if we take the limit $a - b \to 0$,

$$\lim_{a-b\to 0} \det \begin{pmatrix} a & b \\ b & a \end{pmatrix} = 0$$

because the $a - b$ approaches 0. Therefore, the matrix is almost singular and so the system is very ill-condtioned, meaning the system output is sensititve to small changes in coefficients and that small errors in arithmetic will get quickly propagated throughout the calculations. □

1. Apply the bisection routine `bisect` to find the root of the function

$$f(x) = \sqrt{x} - 1.1$$

starting from the interval $[0,2]$ (that is, $a = 0$ and $b = 2$), with `atol` = 1.e-8.

(a) How many iterations are required? Does the iteration count match the expectations, based on our convergence analysis?

(b) What is the resulting absolute error? Could this absolute error be predicted by our convergence analysis?

Solution.

(a) Let us apply the Bisection Method

```
function [root, iter] = bisection_sqrt()
  % Define the function
  f = @(x) sqrt(x) - 1.1;

  % Set the tolerance and initial interval [a, b]
  atol = 1e-8;
  a = 0;
  b = 2; % Initial guess for the root search range

  % Check if the interval is valid
  if f(a) * f(b) > 0
      error('f(a) and f(b) must have opposite signs');
  end

  iter = 0; % Counter for number of iterations

  % Bisection method loop
  while (b - a) / 2 > atol
      iter = iter + 1;
      c = (a + b) / 2; % Midpoint of interval
      if f(c) == 0
          break; % We've found the exact root
      elseif f(a) * f(c) < 0
          b = c; % Root lies in the left subinterval
      else
          a = c; % Root lies in the right subinterval
      end
  end

 root = (a + b) / 2; % Approximate root

 % Display the result
 fprintf('Root found: %.10f\n', root);
 fprintf('Number of iterations: %d\n', iter);
 fprintf('Error: %.10f\n', sqrt(root) - 1.1)
end
```

```
>> amsc460_2
Root found: 1.2100000009
Number of iterations: 27
Error: 0.0000000004
```

27 iterations were required, and this does not match the expectations based on our convergence analysis

$$\mathrm{ceil}(\log_2(b-a)/atol) = 1 \implies 28 \text{ expected iterations}$$

(b) The resulting absolute error is $0.0000000004 < atol$, and this could have been predicted using our convergence analysis.

$\square$

2. Consider the polynomial function[8]

$$
\begin{aligned}
f(x) &= (x-2)^9 \\
&= x^9 - 18x^8 + 144x^7 - 672x^6 + 2016x^5 - 4032x^4 + 5376x^3 - 4608x^2 \\
&\quad + 2304x - 512.
\end{aligned}
$$

   (a) Write a MATLAB script which evaluates this function at 161 equidistant points in the interval $[1.92, 2.08]$ using two methods:

      i. Apply nested evaluation (cf. Example 1.4) for evaluating the polynomial in the expanded form $x^9 - 18x^8 + \cdots$.

      ii. Calculate $(x-2)^9$ directly.

   Plot the results in two separate figures.

   (b) Explain the difference between the two graphs.

   (c) Suppose you were to apply the bisection routine from Section 3.2 to find a root of this function, starting from the interval $[1.92, 2.08]$ and using the nested evaluation method, to an absolute tolerance 10^{-6}. *Without computing anything*, select the correct outcome:

      i. The routine will terminate with a root p satisfying $|p-2| \leq 10^{-6}$.

      ii. The routine will terminate with a root p *not* satisfying $|p-2| \leq 10^{-6}$.

      iii. The routine will not find a root.

   Justify your choice in one short sentence.

Solution. □