Computer Arithmetic & Algebraic Equations

Tutorial homework questions: 4a(i), 8c(ii) due at the beginning of the corresponding tutorial class.

Exercises which you should do by hand (with a calculator, but not a computer) are marked **H**. Exercises in which the use of a computer is central are marked **C**. Exercises illustrating important theoretical points are marked **T**. Optional extra exercises are marked **E**. Advanced optional exercises are marked **A**.

Recommended: H1.b; H2.c,e; HC3.a; HC4.a; C7; H8.b; T9; C11.b; C12.a; CT14.

Round-Off Error and Computer Arithmetic

H1. Compute the absolute error and relative error in approximations of p by p^*.

a. $p = \pi$, $p^* = 3.14$ **b.** $p = \pi$, $p^* = 22/7$ **c.** $p = \pi$, $p^* = 333/106$

d. $p = 137.035999139$, $p^* = 137$. **e.** $p = 0.0072973525664$, $p^* = 1/137$.

H2. Use three-digit (significant figures) rounded arithmetic to perform the following calculations. Compute the absolute error and relative error. (Compute the exact value to at least five digits).

a. $12.3 + 0.499$ **b.** $123 + 0.499$ **c.** 2.07^3 **d.** $(2.07 \times 2.07) \times 2.07$

e. $(17 \div 12) - (24 \div 17)$ **f.** $4.08^2 - 6.9 \times 4.08$ **g.** $(4.08 - 6.9) \times 4.08$

Hint: In Matlab, use `round(x,n,'significant')` or `rnd(x,n)` (from the file `rnd.m` available on the Student Portal) to round x to n significant figures. You can define `r=@(x)round(x,3,'significant'))` so that `r(x)` rounds x to 3 significant figures.

Errors in Scientific Computing

HC3. Use (i) four-digit rounding arithmetic and (ii) single-precision arithmetic to find the roots of the following quadratic equations. Use both forms of the quadratic formula. Compute the absolute and relative errors for these approximations.

a. $1.028x^2 - 12.83x + 0.07316 = 0$

b. $2x^2 - 128x + 1 = 0$

HC4. Evaluate each of the following polynomial at the given point x using (i) decimal arithmetic rounded to n significant figures and (ii) single-precision floating-point arithmetic in Matlab, using both direct evaluation and nested (Horner) form. Compute the errors in each case. Which method performs better?

a. $p(x) = 1.01x^5 - 5.26x^3 - 0.0173x^2 + 0.839x - 1.91, \quad x = 2.44; \quad n = 3.$

b. $p(x) = x^4 - 5.400x^3 + 10.56x^2 - 8.954x + 2.795, \quad x = 1.300; \quad n = 4.$

E5. Suppose the points (x_0, y_0) and $(x_1 y_1)$ are on a straight line with $y_1 \neq y_0$. Three formulae are available to find the x-intercept of the line:

$$x = \frac{x_0 y_1 - x_1 y_0}{y_1 - y_0}, \qquad x = x_0 - \frac{x_1 - x_0}{y_1 - y_0} y_0 \quad \text{and} \quad x = x_1 - \frac{x_1 - x_0}{y_1 - y_0} y_1.$$

Use the data $(x_0, y_0) = (1.36, 3.89)$ and $(x_1, y_1) = (1.44, 4.04)$ and three-digit rounding arithmetic to compute the x-intercept all ways. Which formula is better, and why?

Note: Change in question! Previous version had $(x_1, y_1) = (1.54, 4.34)$.

E6. Use three-digit rounding arithmetic to compute the sum $\sum_{i=1}^{100} 1/i^2$ first by $\frac{1}{1} + \frac{1}{4} + \cdots + \frac{1}{10000}$ and then by $\frac{1}{10000} + \frac{1}{9801} + \cdots + \frac{1}{1}$. Which method is more accurate, and why?

Solutions of Equations of One Variable

C7. Use the built-in Matlab commands `fsolve`, `fzero` and `roots` to find all solutions of the polynomial equation $x^3 + 4x^2 - 9 = 0$. What is the difference between `fsolve` and `fzero`? For what functions can `roots` be used?

Note: Consult the Matlab Help system for information on how to use these commands.

H8. Use (i) the bisection method, (ii) the secant method and (iii) Newton's method to solutions accurate to within (i) 10^{-1} and (ii,iii) 10^{-3} for the following problems:

a. $x^2 - 5 = 0$ on $[2, 3]$.

b. $x^3 - 3x^2 - 7 = 0$ on $[1, 4]$.

c. $e^x = 3x + 4$ on $[0, 4]$.

d. $\sqrt{x} = \cos x$ on $[0, 1]$.

Note: You may use a calcular, but should not use a computer (except to check your answers). Make sure you evaluate cosine in *radians*! There is no need to write out every step in full.

T9. What happens if you apply the bisection method to find the root(s) of $\tan(x) - x$ in $[1, 2]$?
Hint: Plot the functions if you can't see what is going on here!

T10. Use Newton's method to find the roots of the function $f(x) = x^3 - 6x^2 + 7x + 2$ starting at $p_0 = 1$. What happens? Draw the graph of the function and illustrate this graphically.

Find all three roots of the function. Investigate how different starting points yield different roots.

C11. The `bisection_root.m` file (on the Student Portal) defines a function

```
c=bisection_root(f,a,b,e)
```

which computes a root of $f(x) = 0$ in $[a, b]$ using the bisection method, with an error guaranteed to be less than e. At each step, it outputs $[a, c, b,\ f(c), \epsilon = (b - a)/2]$.

Use the `bisection_root` function to estimate the solutions of the following equations to an accuracy of 10^{-6}.

a. $x^3 - 6x^2 + 14x - 7 = 0$ on $[0, 1]$

b. $(x - 3)^2 - \log x = 0$ on $[1, 3]$ and on $[3, 5]$.

c. $e^x - 4x^2 = 0$ on $[0, 2]$ and on $[2, 6]$.

What happens if you apply `bisection_root` to $[1, 5]$ in (b)?

C12. Use the secant method and/or Newton's method to find all solutions of the following equations accurate to within 10^{-5}.

a. $x^4 - 5x^2 + 3x + 1 = 0$

b. $x^2 + 5x \sin(3x) - 1$

Hint: First, try to find an interval containing all the roots. Then roughly locate the roots by looking for changes of sign. Note that a polynomial of degree n has at most n roots.

C13. Write a Matlab function which uses the secant method to estimate a solution of the equation $f(x) = 0$:

```
r=secant(f,p0,p1,eps,nmax)
```

where f is the function, p_0, p_1 are the initial estimates, ϵ is the desired error bound and $n_{\max}$ is a maximum number of steps. The function should return an estimate r of the root.

Write a similar function which uses Newton's method:

```
[r,err]=newton(f,df,p0,nmax,e,d)
```

where df is the derivative of f, p_0 is the initial estimate, e is the desired error bound and d a bound on $f(r)$. The function should return $[r, \epsilon]$, where r is the estimate of the root, and ϵ an estimate of the error.

Test your functions on the problems of Question 11 using a tolerance of 10^{-8}. How many iterates do you require in each case?

CT14. The function $f(x) = \tan(x/2) - 7$ has a unique zero in the interval $[0, 3]$. Use up to 10 iterations of (i) the bisection method, (ii) the secant method and (iii) Newton's method starting at 0.0, 1.5 and 3.0, to approximate this root. Which method is most successful? Sketch a graph of the function, and explain your results.

T15. Apply the bisection method with a tolerance 10^{-1} to attempt to solve the following equation, starting with each of the given intervals:

$$x^3 - 5x^2 + 7x - 2 = 0 \quad \text{on} \quad [0,7],\ [0,5],\ [0,3],\ [1,3],\ [1,2\tfrac{1}{2}],\ [1,2],\ [2,3].$$

What happens in each case? Give all roots of the equations.

E16. Newton's method applied to the function $f(x) = x^2 - a$ with a positive initial approximation p_0 converges to the only positive solution, $\sqrt{a}$.

a. Show that Newton's method in this situation assumes the form that the Babylonians used to approximate square roots:

$$p_{n+1} = \frac{p_n}{2} + \frac{a}{2p_n}.$$

b. Use this formula to estimate $\sqrt{a}$ for values a_i between 0.25 and 4.0, with a spacing of 0.125. Hence plot a graph of the square-root function.

A17. Find a bound for the number of bisection method iterations needed to achieve an approximation accuracy 10^{-3} to the solution of $x^3 + x - 6 = 0$ lying in the interval $[1, 4]$. Find an approximation to the root with this accuracy. How many iterations would you need to find a solution of $e^x + x - 6 = 0$ on the same interval to the same accuracy?

A18. Use (i) the secant method and (ii) Newton's method to find the roots of the function $f(x) = \exp(x) - 3$. Use starting points $p_0 = 0$, $p_1 = 1$ and $p_0 = 2$, $p_1 = 3$. Comment on how the signs of $f(p_n)$ vary. Can you explain this from the graph? What *bounds* on the root do you obtain from the two methods?