Applied Math 5610 **Homework 8** **Due: April 3, 2019**
Bengt Fornberg

1. Show that the entries on row p in Atkinson's Table 6.19 are given in closed form by

$$\beta = 1 / \sum_{i=1}^{p} \frac{1}{i} \quad \text{and} \quad \alpha_j = \frac{(-1)^j \beta}{j+1} \binom{p}{j+1}, \quad j = 0,1,\ldots,p-1.$$

<u>Hint:</u> Set for simplicity $h = 1$. The task becomes then equivalent to approximating $y'(0)$ by a linear combination of the function values $y(-p), y(-p+1), y(-p+2), \ldots, y(-1), y(0)$. Fit a Lagrange interpolation polynomial through this data and evaluate analytically its first derivative at $x = 0$.

2. The best known Runge-Kutta-Fehlberg (RKF45) method is given in Atkinson on page 430. Using six stages, we obtain both a fourth and a fifth order result. The former is accepted as the numerical approximation while the latter is used only for error control. For the fourth order approximation, one can find that the relation between r and ξ is $r = 1 + \xi + \frac{1}{2}\xi^2 + \frac{1}{6}\xi^3 + \frac{1}{24}\xi^4 + \frac{1}{104}\xi^5$.

   a. Plot the stability domain of this RKF45 method.

   b. By setting $r = e^{i\theta}$ and then eliminating r, for ex. Mathematica easily gives the Taylor expansion $\xi(\theta) = i\theta - \frac{1}{780} i\theta^5 - \frac{5}{1872}\theta^6 + \ldots$ Deduce from this:

      i. The scheme is indeed 4th order accurate, and
      ii. The stability domain does not, along the imaginary axis, include any neighborhood of the origin.

3. Consider the linear BVP (boundary value problem)

$$\begin{cases} y'' = 4(y - x) \\ y(0) = 0,\ y(1) = 2. \end{cases}$$

Solve it numerically in the following ways:

   a. Solve two appropriate IVPs (initial value problems) and combine the two solutions to obtain the BVP solution. As your IVP solver, call one of Matlab's built-in routines, such as ode45.

   b. Use finite difference discretization, and solve the resulting linear system. Express your linear system matrix in Matlab's sparse matrix format. Use for example step sizes h = 0.1 and h = 0.05, and carry out one step of Richardson extrapolation.

For both cases (a) and (b), create a figure that graphically displays the error across the interval compared to the exact solution $y(x) = \frac{e^2}{e^4 - 1}(e^{2x} - e^{-2x}) + x.$

4. Consider the nonlinear BVP

$$\begin{cases} y'' = -(y')^2 - y + \ln x \\ y(1) = 0,\ y(2) = \ln 2 \end{cases}.$$

Solve this problem by Newton-based shooting. Start with the initial approximation $y'(1) = s_0$ with $s_0 = 0$ and monitor (display) your convergence. You can again call Matlab's ode45 to solve the IVPs that arise. Compare your converged solution against the analytic solution $y(x) = \ln x$.

<u>Hint:</u> This task can be coded up very cleanly - four Matlab statements together with one "for" and one "end" (in order to loop over the Newton iterations) should suffice.

Additional hints for Problems 3 and 4:

To solve a second order IVP with a solver designed for first order ODEs, rewrite it first as a first order system. For example, $\{y'' = f(x, y, y'),\ y(a) = \alpha,\ y(b) = \beta\}$ becomes after the variable changes $\{y_1(x) = y(x),\ y_2(x) = y'(x)\}$

$$\begin{bmatrix} y_1(x) \\ y_2(x) \end{bmatrix}' = \begin{bmatrix} y_2(x) \\ f(x, y_1(x), y_2(x)) \end{bmatrix}, \quad \text{with initial conditions} \begin{bmatrix} y_1(x) \\ y_2(x) \end{bmatrix}_{x=a} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}.$$

In case of the coupled second order systems that arise in Problem 4, you will need to similarly rewrite this as four coupled first order ODEs.