DUE DATE: NLT 2300 on 18 Nov 2024 (Lesson 34) **VALUE:** 80 Points

SUBMISSION INSTRUCTIONS:

1. You must provide a **documentation statement** with your submission. Per the course letter, missing or insufficient documentation on written work will result in a 20% penalty on the assignment.

2. Your submission must be typed and represent a professional product. *Submitted work should not be a first draft*! Additionally, you must abide by the following **write-up standards:**

- Use only 8 ½ × 11 sheets
- Include a cover page with your name, assignment details, and documentation statement
- Clearly identify each answer (e.g., circle, box, or highlight)
- Problems organized and properly collated

There is a 2-point penalty for each violation of write-up standards. For example, if you use the wrong size paper, do not clearly identify an answer, and rearrange the order of the problems, you will lose 6 points. A violation of the neatness requirement will result in a reduction of up to the entire 80 points for the assignment.

3. Guidelines for each problem write-up:

- *Your write-up should be able to convince a reader as easily as possible that what you have done is correct.* The common argument that "I did it in my head" holds no water unless you know the reader can do it in their head as easily as you did.
- You may assume that the reader has a fair knowledge of the techniques we learn but does not know how to solve the problem – *you must show them how it is done and convince them your method and answer are correct*.
- Be ***complete*** yet ***succinct***. A bloated write-up that includes, for example, mundane algebra computations or scratch work, will likely lose points.

4. **Submission instructions:** Upload your submission to the appropriate assignment in gradescope. You may upload each problem separately or as one file, but you must ensure your problems are neat, legible, and appear in the correct order. Multiple submissions are allowed – only your most recent submission will be graded. Per the course letter, late submissions (without prior coordination AND approval) will result in a loss of 20% or more of the total points on the assignment; work submitted more than 48 hours late (without prior coordination AND approval) will receive no credit.

AUTHORIZED RESOURCES: Anyone and anything with the following exceptions/clarifications:

1. Put in substantial individual effort on a problem before seeking assistance from an outside source. You may then work with other students and/or check a published source and rework the problem if desired. Remember to document any collaboration or source that you reference!

2. *You are not allowed to directly copy any solution*, whether it is from another student or a published source. Do not share any of your written solutions with another student.

3. You must be actively engaged in the process of solving the problem. The work you submit should accurately reflect your understanding of the problem.

PROJECT: Complete all of PARTS 1, 2, and 3 below.

PART 1: Consider the initial value problem (IVP) given below:

$$y'(t) = f\big(t, y(t)\big), \quad a \le t \le b, \quad y(a) = \alpha$$

An *explicit Runge-Kutta (RK) method* is a numerical method that approximates the values $y(t_k)$ at a collection of mesh points $t_k = a + kh$ for $k = 0, \dots, N$ and has the form

$$\begin{aligned} w_0 &= \alpha \\ w_{k+1} &= w_k + \sum_{j=1}^{s} b_j K_j \qquad k = 0, \dots, N-1 \end{aligned}$$

The quantities K_j in the RK method are called *stages* and are computed iteratively:

$$\begin{aligned} K_1 &= hf(t_k, w_k) \\ K_2 &= hf(t_k + c_2 h, w_k + a_{21} K_1) \\ K_3 &= hf(t_k + c_3 h, w_k + a_{31} K_1 + a_{32} K_2) \\ &\vdots \\ K_s &= hf\left(t_k + c_s h, w_k + \sum_{j=1}^{s-1} a_{s,j} K_j\right) \end{aligned}$$

Hence, the RK method is fully determined by the number of stages s and the coefficients a_{ij} (for $0 \le j < i \le s$), b_j (for $1 \le j \le s$), and c_j (for $1 \le j \le s$). The matrix whose $(i,j)^{th}$ value is $a_{i,j}$ is called the *RK matrix*, while the coefficients b_j and c_j are the *weights* and *nodes*, respectively. These parameters are typically arranged in a convenient form known as a *Butcher tableau* (named after mathematician John C. Butcher):

$$\begin{array}{c|ccccc} 0 & & & & & \\ c_2 & a_{21} & & & & \\ c_3 & a_{31} & a_{22} & & & \\ \vdots & \vdots & & \ddots & & \\ c_s & a_{s1} & a_{s2} & \cdots & a_{s,s-1} & \\ \hline & b_1 & b_2 & \cdots & b_{s-1} & b_s \end{array}$$

<u>PROBLEMS:</u>

1. The RK method below is the fourth-order method RK4. Arrange the parameters of RK4 in a Butcher tableau.

$$w_0 = \alpha$$

$$K_1 = hf(t_k, w_k)$$

$$K_2 = hf\left(t_k + \frac{h}{2}, w_k + \frac{1}{2}K_1\right)$$

$$K_3 = hf\left(t_k + \frac{h}{2}, w_k + \frac{1}{2}K_2\right)$$

$$K_3 = hf(t_{k+1}, w_k + K_3)$$

$$w_{k+1} = w_k + \frac{1}{6}(K_1 + 2K_2 + 2K_3 + K_4) \qquad k = 0, \dots, N-1$$

2. The table below contains the Butcher tableaus for three different RK methods. Write out the equations of each RK method, including all stages.

Ralston Method	Heun's Third-Order Method	Runge-Kutta 3/8-Rule
$\begin{array}{c\|cc} 0 & & \\ \frac{2}{3} & \frac{2}{3} & \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$	$\begin{array}{c\|ccc} 0 & & & \\ \frac{1}{3} & \frac{1}{3} & & \\ \frac{2}{3} & 0 & \frac{2}{3} & \\ \hline & \frac{1}{4} & 0 & \frac{3}{4} \end{array}$	$\begin{array}{c\|cccc} 0 & & & & \\ \frac{1}{3} & \frac{1}{3} & & & \\ \frac{2}{3} & \frac{-1}{3} & 1 & & \\ 1 & 1 & -1 & 1 & \\ \hline & \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{array}$

3. For each of the RK methods in problem (2), write a MATLAB script that implements the method. Include as inputs the interval $[a, b]$, the number of mesh points N, the initial condition $y(a) = \alpha$, and the function $f(t, y)$ from the differential equation $y'(t) = f(t, y(t))$. Attach a copy of your code for each method.

<u>PART 2:</u> Once more, consider the IVP given below:

$$y'(t) = f\big(t, y(t)\big), \quad a \le t \le b, \quad y(a) = \alpha$$

If the function $f(t, y(t))$ has rapidly changing behavior for $a \le t \le b$, an RK method may require a very small step size h (i.e., many mesh points $t_k = a + kh$ for $k = 0, \dots, N$) to ensure a desired accuracy in the resulting approximation. In many cases, such a small step size is only needed for t-values on certain parts of the interval $[a, b]$ while a larger step size can be used in areas with less variability. An *adaptive RK method* is designed to optimize the step size at each iteration of a given RK method to save on computational cost.

Suppose the RK method below has global truncation error $\mathcal{O}(h^{p+1})$ for some integer $p \ge 1$

$$\begin{aligned} w_0^* &= \alpha \\ w_{k+1}^* &= w_k^* + \sum_{j=1}^{s} b_j^* K_j \qquad k = 0, \dots, N-1 \end{aligned}$$

Then there exists a second RK method with global truncation error $\mathcal{O}(h^p)$ using the same stages K_j, only differing in the weights b_j

$$\begin{aligned} w_0 &= \alpha \\ w_{k+1} &= w_k + \sum_{j=1}^{s} b_j K_j \qquad k = 0, \dots, N-1 \end{aligned}$$

It can be shown (for details, see §5.5 of the textbook) that the $\mathcal{O}(h^{p+1})$ RK method may be used to estimate the local truncation error of the $\mathcal{O}(h^p)$ RK method at the kth iteration via the relation

$$\tau_{k+1}(h) \approx \tfrac{1}{h} |w_{k+1}^* - w_{k+1}|$$

Hence, computing approximations with both RK methods simultaneously lets one adapt the step size at each iteration to ensure this local truncation error stays below a specified tolerance. If the error is too large, the iteration is repeated with a smaller step size to increase accuracy; if the error is much smaller than the tolerance, the step size is increased for the next iteration to save on computation time.

The augmented Butcher tableau for an adaptive RK method takes the form

$$\begin{array}{c|ccccc}
0 & & & & & \\
c_2 & a_{21} & & & & \\
c_3 & a_{31} & a_{22} & & & \\
\vdots & \vdots & & \ddots & & \\
c_s & a_{s1} & a_{s2} & \cdots & a_{s,s-1} & \\
\hline
 & b_1^* & b_2^* & \cdots & b_{s-1}^* & b_s^* \\
 & b_1 & b_2 & \cdots & b_{s-1} & b_s
\end{array}$$

where the first row of weights b_j^* correspond to the higher order RK method and the second row of weights b_j to the lower order RK method. The RK matrix and nodes are the same for both RK methods.

PROBLEMS:

4. The Runge-Kutta-Fehlberg method (RKF45), which uses two RK methods of orders four and five, is one of the more common implementations of adaptive RK and is given by the following augmented Butcher tableau:

$$
\begin{array}{c|cccccc}
0 & & & & & & \\
\frac{1}{4} & \frac{1}{4} & & & & & \\
\frac{3}{8} & \frac{3}{32} & \frac{9}{32} & & & & \\
\frac{12}{13} & \frac{1932}{2197} & \frac{-7200}{2197} & \frac{7296}{2197} & & & \\
1 & \frac{439}{216} & -8 & \frac{3680}{513} & \frac{-845}{4104} & & \\
\frac{1}{2} & \frac{-8}{27} & 2 & \frac{-3544}{2565} & \frac{1859}{4104} & \frac{-11}{40} & \\
\hline
 & \frac{16}{135} & 0 & \frac{6656}{12825} & \frac{28561}{56430} & \frac{-9}{50} & \frac{2}{55} \\
 & \frac{25}{216} & 0 & \frac{1408}{2565} & \frac{2197}{4104} & \frac{-1}{5} & 0
\end{array}
$$

If the step size at the current iteration of RKF45 is $h > 0$, then the optimal step size needed to achieve a local truncation error of at most $\varepsilon > 0$ is qh where

$$q = \left(\frac{\varepsilon h}{2\left|w_{k+1}^{*} - w_{k+1}\right|}\right)^{1/4} \approx 0.84\left(\frac{\varepsilon h}{\left|w_{k+1}^{*} - w_{k+1}\right|}\right)^{1/4}$$

Write a MATLAB script to implement RKF45, using Algorithm 5.3 in the textbook as a guide. Include as inputs the interval $[a, b]$, the initial condition $y(a) = \alpha$, the tolerance $\varepsilon > 0$, the minimum and maximum allowable step sizes $h_{\min}$ and $h_{\max}$, and the function $f(t, y)$ from the differential equation $y'(t) = f(t, y(t))$. Attach a copy of your code.

<u>PART 3:</u> In the theory of the spread of contagious disease, a relatively simple elementary differential equation can be used to predict the number of infective individuals in the population at any time, under appropriate simplifying assumptions. In particular, assume that all individuals in a fixed population have an equally likely chance of being infected and, once infected, to remain in that state. Suppose $x(t)$ denotes the number of susceptible (non-infective) individuals at time t and $y(t)$ denotes the number of infective individuals at time t. If the population is large enough to assume that both $x(t)$ and $y(t)$ are continuous variables, then the rate at which the number of infective individuals changes is

$$y'(t) = kx(t)y(t) \tag{1}$$

for some proportionality constant k.

<u>PROBLEMS:</u>

5. Assume that the total population is equal to m. Rewrite the differential equation (1) by converting the function $x(t)$ into an expression containing only $y(t)$ and m.

6. Suppose that $m = 100{,}000$, $y(0) = 1000$, and $k = 2 \times 10^{-6}$. Assuming that time is measured in days, find an approximation to the number of infective individuals at the end of 30 days by numerically solving your differential equation from problem (5). Use each of your numerical solvers from problems (2) and (4). For the Ralston Method, Heun's Third-Order Method, and the Runge-Kutta 3/8-Rule use the step sizes $h = 2$ and $h = 1$ to generate two approximations of $y(30)$ each; for RKF45 use minimum and maximum step sizes $h = 1$ and $h = 2$, respectively, to generate one approximation of $y(30)$ having local truncation error $\varepsilon \leq 10^{-2}$. Compare your results for each method. Which method (or methods) do you expect to be the most accurate and why?