Final Project

• As far as length, the final project should be on the same order of magnitude as the last couple of projects you have done (vector class, piecewise interpolation). The difference is that in the previous projects, we wrote part of them in class. In this project it’s your code from start to finish.

• In addition, the final project must be done by only you, not in pairs as was allowed on previous projects.

• You can choose one of the suggested projects or you can email me your idea for a project.

• Your project can provide a class of methods for solving a particular type of problem (e.g., a system of ODEs, a system of nonlinear equations, integrals, etc.) like we have done or it can be application driven (estimating ladybug/aphid populations, estimate world populations, etc.)

• However, your project must incorporate several components.
• Your project must include the following:
  – A written explanation of your project and your program design. This is
    separate from your program and numerical results.
  – Code
  – Numerical results and explanation

• Your code must contain the following components:
  – a module with a definition of a derived type and supporting routines in-
    cluding a constructor and destructor,
  – a main program which tests/uses your module,
  – at least one array (it can be one-dimensional or higher)
  – both functions and subroutines
  – either function overloading or operator overloading
  – at least one do loop and one conditional (can be a select case)
  – one aspect of fortran 90 which we have not used; for example this could
be an intrinsic function which we have not used. The syntax for this command should be given in your write up.

– Our final exam schedule is 7:30-9:30 on Thursday. The projects are due no later than this time.

• In the following slides I will give you some ideas for projects. More than one person can choose the same project but remember that you need to work independently since this counts as your final exam grade.

• The projects I describe are not scripted in the sense that I don’t say for step one do this, etc. Rather they are in the form of general guidelines.
Implementation of Predictor/Corrector Schemes with Variable Stepsize Control

- We have implemented several explicit methods but we did not implemented predictor/corrector schemes.
- You will write code to have routines to perform predictor corrector schemes. You should code schemes through order four and test them out on a couple of problems to make sure you get the appropriate order.
- Compare the work done for a given accuracy to using RK methods.
- After this is complete, you need to incorporate a variable step size control and compare the work done in solving a problem which requires a small time step size in one region but not another with solving the problem using a fixed time step.
- Your results should be illustrated graphically as well as in a table.
In this project we would want to incorporate several of our methods (both single step RK and predictor/corrector) to approximate a system of odes.

You will have to write RK methods through order four for a system and test these on a couple of examples where you can verify the order of convergence.

Then implement predictor/corrector schemes through order four for a system.

The number of equations in your system should be a variable.

Present results both graphically and in tables.

Try your code on a system of ODEs of interest to you; for example an interesting system of 4 ODEs is to start with Newton’s equations of motion for the two-body problem which will define an elliptical orbit; of course you can convert this system of two second order ODEs to a system of four first order ODEs.
Numerical Solution of PDES

• In the last lectures we will introduce finite difference approximations to partial differential equations. If you are already familiar with PDEs, you may choose a project related to this topic.

• For example, you could choose the one (or two-) dimensional heat equation for $u(x, t)$

$$u_t - u_{xx} = f(x, t) \quad a < x < b, t > 0$$

$$u(x, 0) = u_0 \quad u(a, t) = g(t), u(b, t) = h(t)$$

• You could choose a second centered finite difference approximation to $u_{xx}$ and compare using a forward Euler versus a backward Euler for the time derivative term.

• You should approximate your solution for a range of values of $\Delta t, \Delta x$ and compute numerical rates of convergence as well as plotting the results to compare with an exact solution. (To find an exact solution, simply choose a $u(x, t)$ which satisfied the desired boundary and initial conditions, plug it
into the left hand side of your equation and determine $f(x, t)$. 
Numerical Integration/ Numerical Quadrature

• When you took calculus, you quickly discovered that integration is much more difficult than differentiation. In fact, the majority of integrals cannot be integrated analytically.

• When you first learned integration, it was introduced via Riemann sums which were used to approximate the definite integral and then we took the limit as the number of partitions $\to \infty$ to get the definition of the definite integral.

• Consequently, you have already done numerical integration (also called numerical quadrature).

• There are two or three projects you could do on this topic.

• For a function of one independent variable, the basic idea is to replace the definite integral by a sum of the integrand evaluated at certain points (called
quadrature points) multiplied by a number (called quadrature weights).

\[ \int_a^b f(x) \, dx \approx w_1 f(q_1) + w_2 f(q_2) + \cdots + w_n f(q_n) \]

where the \( w_i \) are weights and the \( q_i \) are quadrature points and \( n \) is the number of points.

• For example, for the midpoint rule we have \( n = 1 \), \( q_1 = (a + b)/2 \) and \( w_1 = (b - a) \) to get

\[ \int_a^b f(x) \, dx \approx (b - a) f\left(\frac{a + b}{2}\right) \]

Note that if \( f(x) \geq 0 \) for \( a \leq x \leq b \) then this is just the area of the rectangle with base \( b - a \) and height \( f\left(\frac{a+b}{2}\right) \).

• There are a myriad of other quadrature rules. For example, Simpson’s rule which is in most calculus texts is just

\[ \int_a^b f(x) \, dx \approx \frac{b - a}{6} \left[ f(a) + 4f\left(\frac{a + b}{2}\right) + f(b) \right] \]

• The accuracy of quadrature rules is often gauged by the highest degree polynomial that the rule integrates exactly. For example, the midpoint rule
clearly integrates constant and linear functions exactly. This means that if we integrate $\int_a^b (c_0 + c_1 x) \, dx$ analytically and with the midpoint rule we will get the same answer. However, if we integrate $x^2$ we will not. Simpson’s rule integrates cubic polynomials exactly.

- If the interval of integration is large, then the question arises if a single quadrature rule is good enough.

- One option for this project would be to write routines for the midpoint and Simpson’s rule and then to use them in a composite implementation which basically means we divide the interval $[a, b]$ into subintervals and over each interval we apply one of the rules. If we have a composite Simpson’s rule then over each subinterval of length $h$ we will apply Simpson’s rule.

- Recall that this approach is similar to piecewise interpolation where instead of finding one higher degree polynomial which interpolated all the data, then we pieced together lower order polynomials.

- For this project you would write the midpoint and Simpson’s rule and verify that they integrate the correct order of polynomial and none higher. Then you would incorporate composite rules for both methods and test it out on
several different types of functions. Choose functions that you can integrate exactly over a fixed interval and use this to compute the rate of convergence. For example, if $h$ is the length of your subinterval, then you want to determine $r$ such that the error you make in using your composite rule is $O(h^r)$. For one of your functions plot the exact integral and its approximations as you decrease $h$.

- Notice that in the midpoint rule and Simpson’s rule we used “nice” points, i.e., the endpoints and the midpoint of the interval. As we did in ODEs, one might ask if we are free to choose any point in the interval $[a, b]$ can we choose a point (or points) which give us the highest degree accuracy possible. This yields the so-called Gauss quadrature rules which, to me, are the most useful since we get the most accuracy for our work.

- The only difficulty in the Gauss rules is that they are given on the interval $[-1, 1]$ since they must give you specific points.

- For example, if you look up the one point Gauss rule you will find that it is

$$\int_{-1}^{1} f(x) \, dx \approx 2f(0)$$
which of course is just the midpoint rule. The two point Gauss rule is

\[
\int_{-1}^{1} f(x) \, dx \approx f\left(\frac{-1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right)
\]

where each weight is one here.

- It can be shown that the Gauss rule using \( n \) points is accurate for polynomials of degree \((2n-1)\). For example, the one-point rule is exact for linears and the two point rule is exact for cubics. Compare this with Simpson’s rule which uses three function evaluations to integrate a cubic polynomial exactly.

- If the Gauss rules are only given in terms of the integral from -1 to 1, how can we use them when we want to integrate from \( a \) to \( b \)? We simply do a change of variables and map (linearly) \([-1, 1]\) to \([a, b]\).

- For example, if we take our interval of integration to be \([2, 5]\) and let \( z \) be our independent variable in the integral \( \int_{2}^{5} f(z) \, dz \), then we just have to replace \( x \) in our Gauss rule with the appropriate \( z \) term. We have

\[
z = c_0 + c_1 x, \quad z(-1) = 2, \quad z(1) = 5 \quad \Rightarrow \quad 2 = c_0 - c_1, \quad 5 = c_0 + c_1
\]

\[
\Rightarrow \quad c_0 = \frac{7}{2}, \quad c_1 = \frac{3}{2}, \quad z = \frac{7}{2} + \frac{3}{2} x
\]
Since \( dz = \frac{3}{2} dx \) (i.e., \( dx = \frac{2}{3} dz \)), our integration rule becomes

\[
\int_{-1}^{1} f(x) \, dx \approx 2f(0) \quad \Rightarrow \quad \frac{2}{3} \int_{2}^{5} f(z) \, dz \approx 2f\left(\frac{7}{2}\right) \quad \Rightarrow \quad \int_{2}^{5} f(z) \, dz = 3f\left(\frac{7}{2}\right)
\]

which is exactly the midpoint rule applied to our integral.

- So we have to map our Gauss quadrature points on the interval \([-1, 1]\) to our desired interval \([a, b]\) and multiply our weight by the appropriate factor determined by the relationship between \(dx\) and \(dz\). A formula for this can be computed in terms of \(a, b\) following our example above or looking at another source.

- For this project one could take one of two approaches.

- The first would be to simply code up say one-, two, and three point Gauss rules and use them in a composite form as in the example of Simpson’s and midpoint rule.

- Another option is to code up several of the rules and then to expand them to two dimensions where we are integrating over a box. This is easily done by taking the tensor product of the points. For example, if we have two points
$\pm \frac{1}{\sqrt{3}}$ in $R^1$ then in $R^2$ we have four points given by

\[ \left( -\frac{1}{\sqrt{3}}, -\frac{1}{\sqrt{3}} \right), \quad \left( -\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}} \right), \quad \left( -\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}} \right), \quad \left( \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}} \right) \]

- For the weights we will choose the products of the weights of the corresponding one-d integrations in $x$ and $y$.
- Consequently you could explore integration in one-dimension and two-dimension using Gauss quadrature rules.
Fourier Interpolation

• We considered approximating a complicated function by a polynomial, both a single polynomial and piecewise polynomials.
• We could also approximate functions by trig functions (sines and cosines).
• For example, we could write

\[ f(t) = \frac{a_0}{2} + a_1 \cos t + a_2 \cos 2t + \cdots + a_m \cos mt \]
\[ + b_1 \sin t + b_2 \sin 2t + \cdots + b_m \sin mt \]

• You can consider the case where we are seeking to approximate a complicated function by this trig expansion where we interpolate a given set of points uniformly spaced on some interval.
• In this case there are explicit formulas for the coefficients \( a_i, b_i \).
• For this project you could investigate trigonometric interpolation. Results should be presented in a table as well as graphically.
• Compare your approximation for polynomial interpolation.
We looked at techniques for solving a single nonlinear equation in one independent variable, i.e., finding \( x \) such that \( f(x) = 0 \); e.g., find \( x \) such that \( x^7 - e^x + 6 = 0 \).

We can imagine having a system of nonlinear equations; e.g., three equations in three unknowns

\[
\begin{align*}
f_1(x_1, x_2, x_3) &= 0 \\
f_2(x_1, x_2, x_3) &= 0 \\
f_3(x_1, x_2, x_3) &= 0
\end{align*}
\]

In this project you could implement Newton’s method and some modifications for finding a root of the system.

Comparison of methods and rates of convergence must be performed.

A related problem is that of finding extrema of a function \( f(x_1, x_2, \cdots, x_n) \). Recall that if we want to maximize \( f(x) \) then we find the critical points, i.e.,
where \( f'(x) = 0 \) (plus any boundary points). If we have a function of say three variables then we have to take each partial derivative so we end up with three nonlinear equations to solve.

- You could look at optimizing a function of several variables and consider techniques such as Newton’s method and the Method of Steepest Descent.
Here’s a project that I gave to some Honors Calculus students who had learned to use Mathematica. I will outline it here; more details are available if you choose the topic.

The first goal is the estimate the total number of people who have ever lived on earth for a period of time from a set of data. Start with an estimate for the world population, e.g., starting from 1650 to 2000 in increments of 50 years. Use an integration rule (such as Simpson’s rule over each 100 year period) to integrate this population curve; the result is in person-years. Assume a lifespan of 50 years to estimate number of people who lived. Estimate average population over this period and compare with actual.

There are a couple of simple ODEs to model population growth. For example

\[
\frac{dp}{dt} = \kappa p(t)
\]

where \(p(t)\) represents the population at time \(t\) and \(\kappa\) is the growth factor. Use the data from the table to estimate \(\kappa\). Compare results from numerically
approximating ODE to your result in using the data.

- Use your model to estimate number of people in world at some previous time, e.g., 1492, 1066, etc.

- Another project is predict future populations. To estimate future populations the simple growth model above is not adequate.

- Some scientists (Austin and Brewer) have suggested that we should subtract a growth limitation term from our hyperbolic growth equations. The term will be negative because we want to limit the growth rate \( \frac{dp}{dt} \). The equation we will use is

\[
\frac{dp}{dt} = \frac{k}{p_0} p^2(t) \left( 1 - \frac{p(t)}{p_*} \right).
\]

When \( p(t) = p_* \) then the rate of change of the population is zero and we should have reached the maximum population sustainable on the earth. The value \( p_* \) is called the carrying capacity, i.e., the maximum population the earth is able to sustain. This differential equation is much more difficult to solve than our previous one. In fact, *Mathematica* is unable to solve it. However, one can show that the solution of this equation satisfies the
nonlinear equation

\[ kt - \left(1 - \frac{p_0}{p(t)}\right) - \frac{p_0}{p_*} \ln \left[ \left(\frac{p_* - p_0}{p_0}\right) \left(\frac{p(t)}{p_* - p(t)}\right) \right] = 0. \]

Ideally we would have liked to have an equation saying \( p(t) = \cdots \) but this is not attainable so we will have to work with this equation. If we have values for the constants \( p_* \), \( k \), and \( p_0 \) then for a fixed time \( t \) we could solve this nonlinear equation for \( p(t) \). We have studied routines for root finding.

- Use the data from the table to estimate constants; estimate population at future times. Determine when population reaches its maximum value. How many additional people will be alive in 2003 than in 2002? This increase in the population is equivalent to adding an entirely new country. Determine where this country would rank among the countries of the world. You can use the population statistics from 2000. You can find a ranking of countries by population on the web at

http://www.census.gov/ipc/www/idbnew.html
Vermeer Forgeries

After the liberation of Belgium in World War II, the Dutch Field Security began its hunt for Nazi collaborators. They discovered, in the records of a firm which had sold numerous works of art to the Germans, the name of a banker who had acted as an intermediary in the sale to Goering of the painting *Woman Taken in Adultery* by the famed 17th century Dutch painter Jan Vermeer. The banker in turn revealed that he was acting on behalf of a third-rate Dutch painter H.A. Van Meegeren, and on May 29, 1945 Van Meegeren was arrested on the charge of collaborating with the enemy. On July 12, 1945 Van Meegeren startled the world by announcing from his prison cell that he had never sold *Woman Taken in Adultery* to Goering. Moreover, he stated that this painting and the very famous *Disciples at Emmaus* as well as four other presumed Vermeers and two de Hooghs were his own works. Many people thought that Van Meegeren was lying to save himself from the charge of treason. To prove his point, Van Meegern began, while in prison, to forge the Vermeer painting *Jesus Amongst the Doctors* to demonstrate to the skeptics just how good a forger of Vermeer
he really was. The work was nearly completed when Van Meegeren learned that a charge of forgery had been substituted for that of collaboration. He therefore refused to finish and age the painting in the hope that investigators would not uncover his secret of aging his forgeries. To settle the question, an international panel of distinguished chemists, physicists and art historians was appointed to investigate the matter. The panel took x-rays of the paintings to determine whether other paintings were underneath them. In addition, they analyzed the pigments (coloring materials) used in the paint and examined the paintings for certain signs of age. Van Meegeren was well aware of these methods. To avoid detection, he scraped the paint from old paintings that were not worth much just to get the canvas, and he tried to use pigments that Vermeer would have used. Van Meegeren also knew that old paint was extremely hard and impossible to dissolve. Therefore, he cleverly mixed a chemical (phenoformaldehyde) into his paint, and this hardened into Bakelite when the finished painting was heated in an oven. However, Van Meegeren was careless with several of his forgeries, and the panel of experts found traces of the modern pigment cobalt blue. In addition, they also detected the phenoformaldehyde (which was first discovered at the close of the 19th century) in several of the paintings. On the basis of the evidence Van Meegeren was convicted on October 12, 1947 and sentenced to one
year in prison. While in prison he suffered a heart attack and died on December 30, 1947. Despite the evidence gathered by the panel of experts, many people still refused to believe that the famed *Disciples at Emmaus* was forged by Van Meegeren. Their contention was based on the fact that the other alleged forgeries and Van Meegeren's nearly completed *Jesus Amongst the Doctors* were of a very inferior quality. Surely, they said, the creator of the beautiful *Disciples at Emmaus* could not produce such inferior pictures. Indeed, the *Disciples at Emmaus* was certified as an authentic Vermeer by the noted art historian A. Bredius and was bought by the Rembrandt Society. The answer of the panel to these skeptics was that because Van Meegeren was keenly disappointed by his lack of status in the art world, he worked on the *Disciples at Emmaus* with the fierce determination of proving that he was better than a third-rate painter. After producing such a masterpiece his determination was gone. Moreover, after seeing how easy it was to dispose of the *Disciples at Emmaus* he devoted less effort to his subsequent forgeries. This explanation failed to satisfy the skeptics. They demanded a thoroughly scientific and conclusive proof that the *Disciples at Emmaus* was indeed a forgery. This was done in 1967 by scientists at Carnegie Mellon University.
The key to the dating of paintings and other material such as rocks and fossils lies in the phenomenon of radioactivity discovered at the turn of the century. In this project one with look at standard IVPs for estimating the amount of radioactive material present in a sample at a particular time and comparing it with those measured in several “Vermeer” paintings.
Predator-Prey Models

- There are numerous ODE models for predator-prey. These can model, e.g., ladybug/aphid populations in the garden, wolf population in Yellowstone, etc.

- For example, a simple system for the ladybug/aphid population is

\[
\frac{dA}{dt} = A(1.0 - L)
\]

\[
\frac{dL}{dt} = L(A - 0.2)
\]

where \(A(t)\) represents the aphid population in millions at time \(t\) and \(L(t)\) denotes the ladybug population in millions at time \(t\) in days.

- If a pesticide is used in addition, then the model could become

\[
\frac{dA}{dt} = A(1.0 - L) - 2.5sA
\]
\[
\frac{dL}{dt} = L(A - 0.2) - 0.1sL
\]

where \( s \geq 0 \) denotes the strength of the pesticide with \( s = 1 \) being full strength.

- This project involves implementing methods such as predictor-corrector techniques for a particular application. Specifics available upon request.