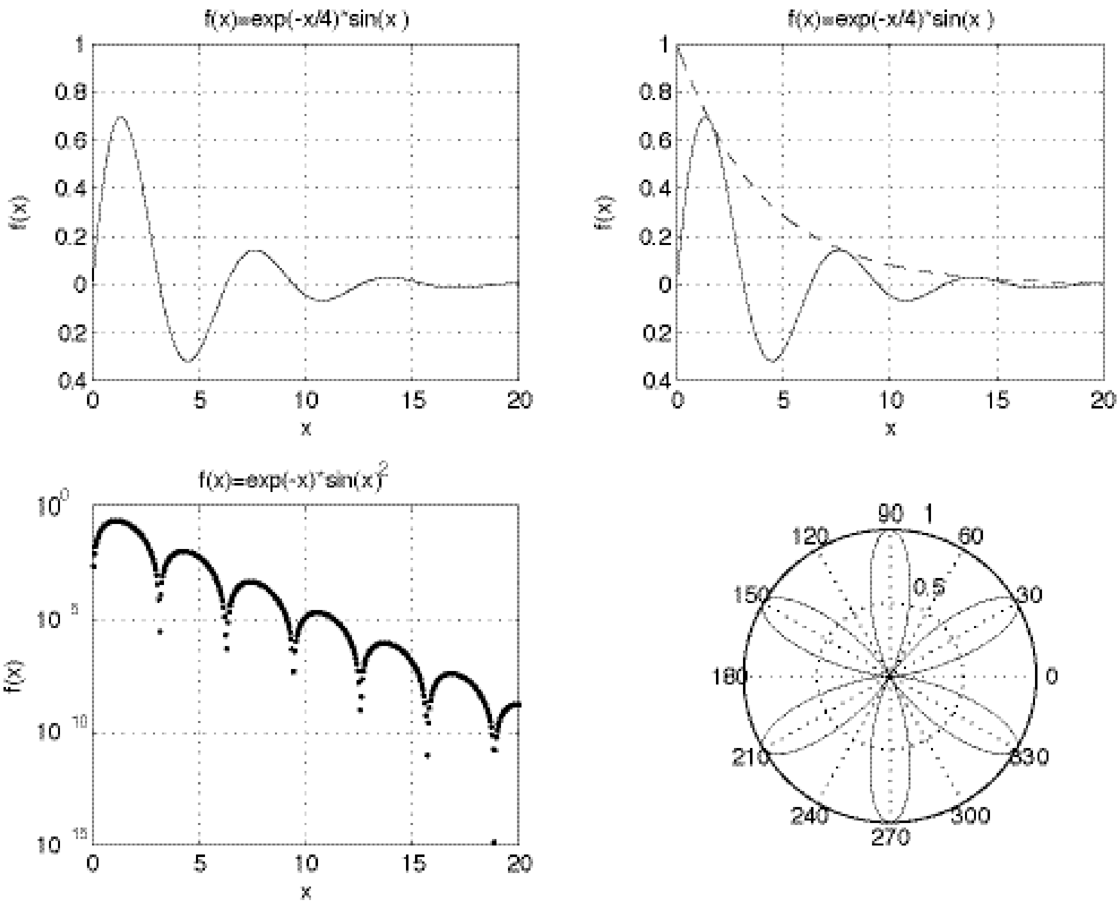
PHYS 416 - Problems from Chapter 1

- *Numbered questions are as in the Garcia book; lettered questions are* ***not*** *from the book.*
- *When you turn in your programs, please use the same numbering convention, for example the programs for first problem below should be numbered ex2a.m, ex2b.m, etc for parts (a), (b) and so on.*

MATLAB 'Warm-up' Exercises

[due by the start of class on January 15]

3. Reproduce the following graph (on 1 page). Try to be as accurate as possible in your reconstruction.



Programming exercises

A. *The hailstone or 3n+1 problem.* Let n be a positive integer; we iterate n with the following procedure: if n is even, we divide n by 2, if n is odd we replace n by 3n+1. In summary:

$$n = \begin{cases} \dfrac{n}{2} & \text{for n even} \\ 3n+1 & \text{for n odd} \end{cases}$$

The algorithm continues until $n = 1$. It is not known whether the above process terminates at 1 for any starting point, but no number has yet been found that does not terminate at 1, however no general proof exists (The Collatz conjecture).

Write a program that inputs a value of n and iterates until n becomes 1. Plot the computed number versus iteration. [Hint: Use the while statement.]

B. Now modify your program so that it computes an array of the array of numbers as a function of iteration. In other words construct a matrix f(N,k) which stores the computed number (n) as a function of starting number N at iteration k and g(N) which is the number of iterations needed to get to the number 1. For example, if N=3, f would look like:

$$f = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 10 & 5 & 16 & 8 & 4 & 2 & 1 \end{bmatrix} \qquad g = \begin{bmatrix} 0 \\ 1 \\ 7 \end{bmatrix}$$

(A 0 means there are no values, and acts as a filler for the matrix). Make a scatter plot of these functions (in separate windows), and make its axis range from 1 to N. Use the number range from 1 to about 200. (You can try a larger upper number, but it will take considerably longer and the resulting plots will be rather full.)

14. Modify **orthog.m** so that if the second vector is not orthogonal to the first, the program computes a new vector that is orthogonal to the first vector, has the same length as the second vector, and is in the same plane as the two input vectors. This orthogonalization is often used with eigenvectors and is commonly performed using the Gram-Schmidt procedure. [Computer]

15. From the tables we find the following values for the zeroth-order Bessel function: $J_0(0) = 1$ $J_0(0.5) = 0.9385$; $J_0(1.0) = 0.7652$. Using **intrp.m**, find the estimated values of $J_0(x)$ for range $x = 0.3, 0.9, 1.1, 1.5$ and 2.0. Use the intrinsic matlab functions for the Bessel function to compute the tabulated values and compare. [Computer]

16. Modify **intrpf** so that it can handle any number of data points by using higher-order polynomials. After testing your program, give it the following values of the Bessel function: $J_0(0) = 1.0$; $J_0(0.2) = 0.9900$; $J_0(0.4) = 0.9604$; $J_0(0.6) = 0.9120$; $J_0(0.8) = 0.8463$; $J_0(1.0) = 0.7652$, and repeat the previous exercise. Do your estimates improve? [Computer]

18. A Bézier cubic curve is defined by the parametric equations

$$x(t) = a_x t^3 + b_x t^2 + c_x t + x_1$$

$$y(t) = a_y t^3 + b_y t^2 + c_y t + y_1$$

where $0 \le t \le 1$. The Bézier control points are given by the relations)

$$x_2 = x_1 + c_x/3 \qquad y_2 = y_1 + c_y/3$$

$$x_3 = x_2 + (c_x + b_x)/3 \qquad y_3 = y_2 + (c_y + b_y)/3$$

$$x_4 = x_1 + c_x + b_x + a_x \qquad y_4 = y_1 + c_y + b_y + a_y$$

The curve goes from $(x(0),y(0))=(x_1,y_1)$ to $(x(1),y(1))=(x_4,y_4)$and is tangent to the lines (x_1,y_1)–(x_2,y_2) and (x_3,y_3)–(x_4,y_4). Write a program to draw a Bézier curve, given the control points (x_1, y_1),…..,(x_4,y_4). Draw the curve with the control points (0,0), (2,1), (-1,1), (1,0). [Computer]

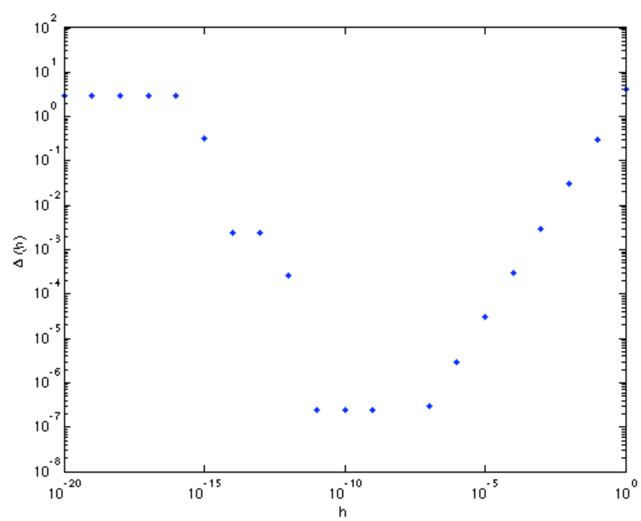
22. The double factorial is defined as

$$n!! \equiv n \times (n-2) \times (n-4) \ldots \times \begin{cases} 6 \times 4 \times 2 & n \text{ even} \\ 5 \times 3 \times 1 & n \text{ odd} \end{cases}$$

(a) Write a program that prints out $n!!$ by evaluating its definition using logarithms. Test your program by checking that $1000!! \approx 3.99 \times 10^{1284}$; compute 2001!!.

(b) Obtain an expression for $n!!$ in terms of $n!$. [Pencil]

24. (a) Write a program to reproduce similar to Figure 1.3 of the text (shown below) but for $f(x)=x^3$ at x=1. Use h = 1, 10^{-1}, 10^{-2},…



C. For the following sequence

$$x_2 = 2,\ x_{n+1} = 2^{n-1/2}\sqrt{1-\sqrt{1-4^{1-n}x_n^2}}\ , n = 2,3,....$$

it can be shown that $\lim_{n\to\infty} x_n \to \pi$. Write a program the plots the relative difference between x_n and π in the form:

$$e_b = \frac{|\pi - x_n|}{\pi}$$

as a function of n in the range [2,30]. Use a log scale for e_n. What do you find? Explain why. [Computer]

D. Write a program that plots the polynomial $y(x) = (x-1)^n$ as well as the expanded binomial expansion $y(x) = \sum_{k=0}^{n} \frac{n!}{m!(n-k)!} x^{n-k}(-1)^k$. Plot the two versions of y(x) in the range x=0.96 to 1.04 for n=5 and n=9 and comment on the results. [Computer]