Assignment #8

DUE *Friday 4 December, 2009 at 5pm*

Problems 6.4, exercise 14.

Problems 6.4, exercise 17. (*Do this by hand. You may use the program* `ncspline.m`, *which is online at* `http://www.dms.uaf.edu/~bueler/ncspline.m`, to *check* your answer, but I encourage you to do this easy problem by returning to the meaning of natural cubic splines. Namely, that they have continuous of first and second derivatives at interior nodes and $S'' = 0$ at end nodes.)

Computer Problems 6.4, exercise 1. Concretely, use 11 points and compute the cubic splines with `ncspline.m`.

Problems 7.2, exercise 1.

Problems 7.2, exercise 14.

Problems 7.2, exercise 31.

Exercise 1. In lecture, while deriving Simpson's rule and the error term for trapezoid rule, I claimed certain integrals came out certain ways. Please fill in these easy calculations:

(a) On the interval $[a, b]$ and with $x_0 = a$, $x_1 = (a + b)/2$, $x_2 = b$, compute Lagrange polynomials $\ell_0(x)$, $\ell_1(x)$, $\ell_2(x)$. Then show

$$A_0 = \int_a^b \ell_0(x)\, dx = \frac{b-a}{6},$$
$$A_1 = \int_a^b \ell_1(x)\, dx = \frac{4(b-a)}{6},$$
$$A_2 = \int_a^b \ell_2(x)\, dx = \frac{b-a}{6}.$$

Describe in a sentence what these quantities are.

(b) Show

$$\int_a^b (x-a)(b-x)\, dx = \frac{(b-a)^3}{6}.$$

Describe in a sentence what role this quantity played.

Exercise 2. Using the error term on page 484, find what n is needed to approximate

$$\int_0^{10} \pi + \sin(3x)\, dx$$

to within 10^{-8} using composite Simpson's rule. Then apply composite Simpson's rule, and show, by computing the exact integral, that the actual error is that small.

Exercise 3. Write a *quadratic* spline program. In particular, given $a = t_0 < t_1 < \cdots < t_n = b$ and $y_0, y_1, \dots, y_n$, the program will evaluate $Q(x)$ at given x values in $[a, b]$. Here

$$Q(x) = Q_j(x) = a_j + b_j(x - t_j) + c_j(x - t_j)^2 \qquad \text{on} \quad t_j \le x \le t_{j+1},$$

with

$$\begin{aligned} Q_j(t_j) &= y_j, & j &= 0, 1, \dots, n-1, \\ Q_j(t_{j+1}) &= y_{j+1}, & j &= 0, 1, \dots, n-1, \\ Q_j'(t_{j+1}) &= Q_{j+1}'(t_{j+1}), & j &= 0, 1, \dots, n-2. \end{aligned}$$

Show that we still need one more condition. You may use the additional condition $c_0 = 0$.

You will discover that you can solve the system "by hand" for all the coefficients, because the matrix you need to solve is bi-diagonal and lower triangular (at least the way I set it up). Comment on the geometric and algorithmic consequences of the choice $c_0 = 0$, and suggest better choices.

The inputs and outputs of your code will be like `ncspline.m`, which is online. In fact, I encourage you to base your work on that code, but the core of your code can be simpler than `ncspline.m` because you do not need to set up and solve a system, but you can replace that part with a loop which computes coefficients.