CS 370 Winter 2013: Assignment 1

Instructor: Yuying Li	Office: DC3623	e-mail: *yuying@uwaterloo.ca*
Lec 001 :	MWF 8:30-9:20am MC2054	OH (Li): Tues 2-3pm
TAs:		
Nian Ke	knian@uwaterloo.ca	DC3594
Mark Prosser	rmprosse@uwaterloo.ca	DC2302B
Haofan Zhang	h267zhan@uwaterloo.ca	DC3594

Web Site: www.student.cs.uwaterloo.ca/~cs370/

Due Feb 1, 2013, 5:00 PM, in the Assignment Boxes, 3rd Floor MC. Please attach a cover page, which you can download at the course website, to your submitted assignment.

1. **(7 marks)** Let $F(2, 53, -1022, 1023)$ denote IEEE double precision floating system.

   (a) What are the smallest and largest positive numbers in F?

   (b) How many numbers are in F?

   (c) What fraction of the numbers of F are in the interval $1 \leq x < 2$? What is the distance between two consecutive numbers of F in the interval $1 \leq x < 2$?

   (d) What fraction of the numbers of F are in the interval $\frac{1}{64} \leq x < \frac{1}{32}$? What is the distance between two consecutive numbers of F in the interval $\frac{1}{64} \leq x < \frac{1}{32}$?

2. **(10 marks)** Carry out (by hand, with the aid of a calculator) the following computations by simulating the 5-digit rounding arithmetic of the floating point number system $F(10, 5, -10, 10)$.

   (a) The roots r_1 and r_2 of the quadratic equation $ax^2+bx+c = 0$ are given by the following well-known formulas:
   $$r_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}, \qquad r_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$
   when $a \neq 0$. Calculate the roots r_1 and r_2 using arithmetic in $F(10, 5, -10, 10)$ for the quadratic equation
   $$x^2 + 111.11x + 1.2121 = 0. \tag{1}$$
   Compare the computed results with the true roots (to 5 significant digits) which you may calculate on a computer using 10 or more digits of precision. Specifically, what is the relative error in r_1 and in r_2?

   (b) Note that a *cancellation problem* arises when applying the above formulas for any quadratic equation having the property that
   $$|b| \approx \sqrt{b^2 - 4ac}.$$
   For an equation with this property, if $b > 0$ then the quadratic formula for r_1 will exhibit cancelation, and if $b < 0$ then r_2 will exhibit cancelation. Show that a mathematically equivalent formula for r_1 is
   $$r_1 = \frac{2c}{-b - \sqrt{b^2 - 4ac}}.$$
   Hint: Rationalize the numerator (i.e. multiply the numerator and denominator of the original formula for r_1 by an appropriate quantity).

   (c) The formula for r_2 can be manipulated in a similar manner. Deduce a better algorithm for calculating the roots of a quadratic equation and present it in the following form.

Algorithm R.

if $b > 0$ then
$\quad r_2 = \frac{-b-\sqrt{b^2-4ac}}{2a}$
$\quad r_1 = \frac{c}{a\ r_2}$
else
$\quad r_1 =$
$\quad r_2 =$
end

(d) Redo the calculation of the roots of equation (1) by applying Algorithm R using arithmetic in the same floating point number system. Compare the computed results with the true roots. Specifically, what is the relative error in r_1 and in r_2? How do the computed results of part (d) compare with the computed results of part (a)?

(e) Redo the calculation of the roots of equation (1) by applying your improved algorithm, using the same FPNS $F(10, 4, -10, 10)$. Compute the relative errors of the computed roots. Compare the accuracy of the computed roots from (d) with the results from (a).

3. **(5 marks)** A recurrence sequence $\{p_n\}$ defined by

$$p_n = \frac{50}{7} p_{n-1} - p_{n-2}, \quad n \geq 2$$

has as its exact solution

$$p_n = a \cdot 7^n + b \cdot 7^{-n}$$

where $a = \frac{1}{48}(7p_1 - p_0)$ and $b = \frac{7}{48}(7p_0 - p_1)$. Thus if we use

$$p_0 = 1 \ ; \quad p_1 = \frac{1}{7} \ ;$$

with exact arithmetic then $p_n = \frac{1}{7^n}$ for all $n \geq 0$.

Suppose now that we compute this sequence using the same starting values but now computing with floating point arithmetic. Carry out a stability analysis for this computation by considering the propagation of errors introduced in the initial values p_0 and p_1. Is this computation stable or unstable? (**Note:** A Matlab experiment is not an acceptable solution for this question.)

4. **(4 marks)** The financial derivative digital option has the following payoff function

$$f(S) = \begin{cases} 1 & \text{if } S > 100 \\ 0 & \text{otherwise} \end{cases}$$

In order to valuate this option, we may want to smooth the payoff function $f(S)$. Noting that $f(S)$ is discontinuous at $S = 100$, we first revise the definition of the payoff as

$$\hat{f}(S) = \begin{cases} 1 & \text{if } S > 100 \\ 0.5 & \text{if } S = 100 \\ 0 & \text{if } S < 100 \end{cases}$$

(a) Show that the following piecewise quadratic $q(S)$

$$q(S) = \begin{cases} 0 & \text{if } S \leq 99.9 \\ 50(S - 99.9)^2 & \text{if } 99.9 \leq S \leq 100 \\ 1 - 50(S - 100.1)^2 & \text{if } 100 \leq S \leq 100.1 \\ 1 & \text{if } S \geq 100.1 \end{cases}$$

is a quadratic interpolating spline for $\hat{f}(S)$ at $S_1 = 99.9$, $S_2 = 100$, $S_3 = 100.1$ with the boundary conditions

$$q'(S_1) = \hat{f}'(S_1), \; q'(S_3) = \hat{f}'(S_3)$$

(b) In one plot, graph the original digital payoff $f(S)$ and the quadratic spline approximation $q(S)$ in the interval $[99.5, 100.5]$.

5. **(6 marks)** For data points $(x_1, y_1), \cdots, (x_N, y_N)$, $x_1 < x_2 < \cdots < x_N$ and $N \geq 4$, not-a-knot cubic spline has the end condition that the third order derivative is continuous at x_2 and x_{N-1}. Assume that $S(x)$ is a cubic spline interplant for the four data points $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$

$$S(x) = \begin{cases} p_1(x) & x \in [x_1, x_2] \\ p_2(x) & x \in [x_2, x_3] \\ p_3(x) & x \in [x_3, x_4] \end{cases}$$

Suppose $P(x) = x^3 - 1$ is the cubic interplant for the same four points $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$ where $x_1 < x_2 < x_3 < x_4$ are knots. What is the not-a-knot spline interplant $S(x)$? Provide a detailed proof for your answer.

6. **(16 marks)** Create parametric curves representing the outline of your hand and your first and last initials separated by an ampersand in handwriting inside the hand. For example, John Smith would produce the writing

$$J \;\&\; S$$

inside the outline of your hand.

Follow the steps below:

(a) Start with the following script file named **init.m** to initialize data arrays

```
figure('position',get(0,'screensize'))
axes('position',[0 0 1 1])
[x,y]=ginput;
⋮
v=axis;
clf;
```

Use **help** command to learn any matlab command that is unfamiliar to you. You can trace your hand and initials first on a piece of paper and then put the paper on the computer screen. Create your data for both your hand and initial using the mouse to select a few dozen points outlining your hand and initials. Terminate each input sequence with a carriage return.

(b) Using the data arrays from **init.m**, generate parametric curve representations based on smooth parametric curve interpolations as described in your course notes. At least one smooth segment of the image should be curved enough to be interesting (change your initials temporarily if need be). Show the output using piecewise linear splines and cubic splines. More precisely you are to do the following.

Prepare three Matlab.m files, one for each of the following tasks. *Use the same axis scaling v from* **init.m** *for subsequent plots.*

(1) A plot with grid lines and the interpolation data corresponding to the crude initial shape plotted with the symbol '*'. Plot axis and grid lines for these plots. The plot should have a title.

(2) A plot of your letters and the hand created by joining the original data points with straight lines. (The letters will not look very smooth). Plot axis and grid lines for this plot. The plot should have a title.

(3) A smooth plot of your hand and initial created by refining the parameter by a factor of 3. Use different spline end conditions (see matlab functions *csape* and *fnval*) for the letters and the hand. For example, if the first uses not-a-knot conditions, you might try natural conditions for the second.

Plot this without grid or axes. The plots should have titles.

Submit all plots and hard copies of your codes.