Marking Sheet Assignment 1 CS370 Winter 2013

Name: .... ID:................

1.	(a) smallest, largest	(2)	__________	
	(b) number of numbers	(1)	__________	
	(c) fraction, distance	(2)	__________	
	(d) fraction, distance	(2)	__________	
				......./7
2.	(a) r1, r2, errors	(2)	__________	
	(b) new r1	(2)	__________	
	(c) improved algorithm	(2)	__________	
	(d) improved r1, r2, errors	(2)	__________	
	(e) improved r1, r2, errors	(2)	__________	
				......./10
3.	stability analysis			
				......../5
4.	(a) quadratic interpolating spline	(2)	__________	
	(b) plot	(2)	__________	
				......../4
5.	not-a-knot spline			
				......../6
6.	(1) plots (2), code (2)	(4)	__________	
	(2) plots (2), code (2)	(4)	__________	
	(3) plots (2), code (6)	(8)	__________	
				......../16

Total: (48) ______________

1. (a) The smallest positive number is 0.1×2^{-1022}; the largest is $0.\underbrace{1\ldots1}_{53 one's} \times 2^{1023}$ ■

   (b) F has

$$|F| = \underbrace{1}_{\text{for } 0} + \underbrace{1}_{\text{leading one}} \times \underbrace{2^{52}}_{\text{52 remaining digits}} \times \underbrace{(1022 + 1023)}_{\text{exponents of 2}}$$
$$= 1 + 2025(2^{52})$$

■

   (c) Rewrite the inequality:

$$1 \le x < 2 \iff 0.1 \times 2^1 \le x < 0.1 \times 2^2$$
$$\text{Fraction} = \frac{\underbrace{1}_{\text{leading one}} \times \underbrace{2^{52}}_{\text{52 remaining digits}} \times \underbrace{1}_{\text{one exponent}}}{|F|} \approx 0.0004938$$

   The distance for consecutive numbers is $1 \times 2^{-53} \times 2 = 2^{-52}$ ■

   (d) Rewrite the inequality:

$$\frac{1}{64} \le x < \frac{1}{32} \iff \frac{1}{2^6} \le x < \frac{1}{2^5} \iff 0.1 \times 2^{-5} \le x < 0.1 \times 2^{-4}$$
$$\text{Fraction} = \frac{1 \times 2^{52} \times 1}{|F|} \approx 0.0004938$$

   The distance for consecutive numbers is $1 \times 2^{-53} \times 2^{-5} = 2^{-58}$ ■

2. (a) First caculate the parts:

$$b^2 = 12345.4321 \approx 12345$$
$$4ac = 4 \times 1 \times 1.2121 = 4.8484$$
$$b^2 - 4ac = 12340.1516 \approx 12340$$
$$\sqrt{b^2 - 4ac} = 111.085552615990527825 \cdots \approx 111.09$$
$$-b + \sqrt{b^2 - 4ac} = -0.02$$
$$-b - \sqrt{b^2 - 4ac} = -222.2$$

   Hence, $(fl(r_1), fl(r_2)) \approx (-0.01, -111.1)$. By calculator,
   $(r_1, r_2) \approx (-0.0109100803694867128, -111.099089919630513287)$.
   Hence, the errors are $(\delta_{r_1}, \delta_{r_2}) \approx (\ 0.083416, -8.1916 \times 10^{-6})$. ■

(b) Rationalize the numerator:

$$\begin{aligned}
r_1 &= \frac{\sqrt{b^2-4ac}) - b}{2a} \\
&= \frac{(\sqrt{b^2-4ac}) - b) \times (\sqrt{b^2-4ac} + b)}{2a \times (\sqrt{b^2-4ac} + b)} \\
&= \frac{b^2 - 4ac - b^2}{2a \times (\sqrt{b^2-4ac} + b)} \\
&= \frac{-2c}{\sqrt{b^2-4ac} + b} \\
&= \frac{2c}{-b - \sqrt{b^2-4ac}}
\end{aligned}$$

As required. ■

(c) Derive r_2 in term of r_1:

$$\begin{aligned}
r_2 &= \frac{-b - \sqrt{b^2-4ac}}{2a} \\
&= \frac{-(b + \sqrt{b^2-4ac})}{2a} \\
\implies &= \frac{-(b^2 - b^2 - 4ac)}{2a \times (b - \sqrt{b^2-4ac})} \\
&= \frac{2c}{b - \sqrt{b^2-4ac}} \\
\implies &= \frac{-2c}{-b + \sqrt{b^2-4ac}} \\
\implies &= \frac{-2c}{\frac{-b+\sqrt{b^2-4ac}}{2a} \times 2a} \\
\implies &= \frac{-c}{ar_1}
\end{aligned}$$

Algorithm R.
if $b > 0$ then

$$r_2 = \frac{-b - \sqrt{b^2-4ac}}{2a}$$
$$r_1 = \frac{c}{ar_2}$$

else

$$r_1 = \frac{\sqrt{b^2-4ac}) - b}{2a}$$
$$r_2 = \frac{-c}{ar_1}$$

end ■

(d)

$$r_2 = -111.1 \text{ (from before)}$$
$$r_1 = \frac{1.2121}{1 \times -111.1} = -0.010910$$

Hence, the errors $(\delta_{r_1}, \delta_{r_2}) \approx (7.3665 \times 10^{-6}, -8.1916 \times 10^{-6})$. Unlike (a), the errors for (d) are close to 0. Hence, Algorithm R is a better algorithm than the one used in (a). ■

(e)

$$b^2 = 111.1^2 = 12343.21 \approx 12340$$
$$4ac = 4 \times 1 \times 1.212 \approx 4.848$$
$$b^2 - 4ac = 12335.152 \approx 12340$$
$$\sqrt{b^2 - 4ac} = 111.0855526 \cdots \approx 111.1$$
$$-b - \sqrt{b^2 - 4ac} = -222.2$$
$$r_2 = -111.1$$
$$r_1 = \frac{1.212}{1 \times -111.1} = -0.01090909 \cdots \approx -0.01090$$

Hence, the errors $(\delta_{r_1}, \delta_{r_2}) \approx (9.2395 \times 10^{-4}, -8.1916 \times 10^{-6})$. Unlike (a), the errors for (d) are close to 0. Hence, Algorithm R is a better algorithm than the one used in (a). ■

3. Take the limit of p_n:

$$\begin{aligned} \lim_{n\to\infty} p_n &= a \lim_{n\to\infty} 7^n + b \lim_{n\to\infty} \frac{1}{7^n} \\ &= a \lim_{n\to\infty} 7^n + 0 \end{aligned}$$

But a, which is a constant based on p_0 and p_1, $a = \frac{1}{48}(7(\frac{1}{7}) - 1) = 0$. Hence the limit goes to zero. Therefore, the given computation is stable. ■

4. (a) Check whether $q(S) = \hat{f}(S)$ at $S1, S2, S3$, and continuity:

$$q(99.9) = 50(99.9 - 99.9)^2 = \hat{f}(99.9) = 0$$
$$q(100) = 50(100 - 99.9)^2 = 1 - 50(100 - 100.1)^2 = \hat{f}(100) = 0.5$$
$$q(100.1) = 1 - 50(100.1 - 100.1)^2 = \hat{f}(100.1) = 1$$

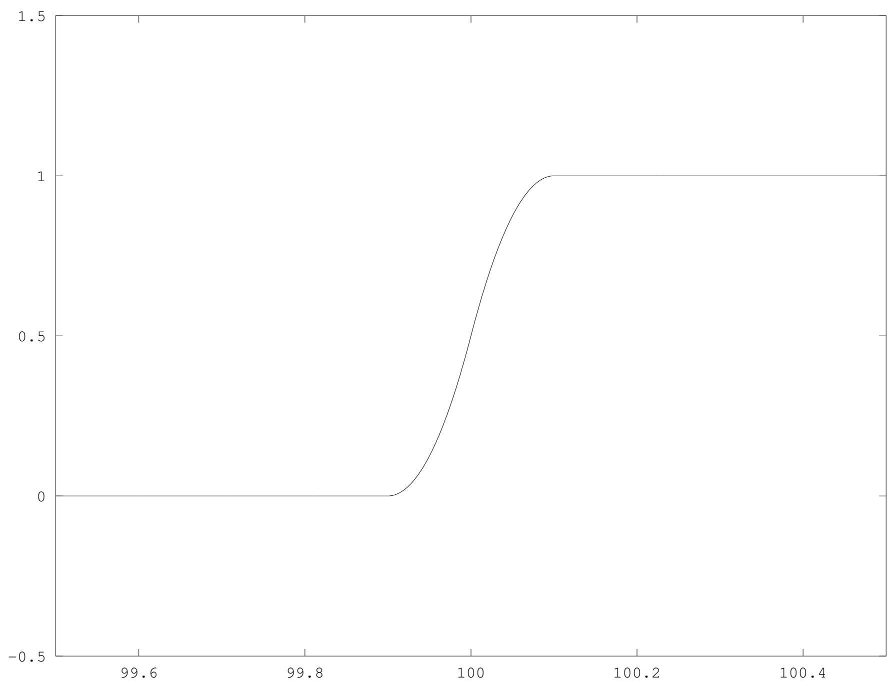
Hence q is continuous at $S1, S2, S3$ and contains the break-points. Now check the continuity of q':

$$q'(S) = \begin{cases} 0 & \text{if } S \leq 99.9 \\ 100(S - 99.9) & \text{if } 99.9 \leq S \leq 100 \\ -100(S - 100.1) & \text{if } 100 \leq S \leq 100.1 \\ 0 & \text{if } S \leq 99.9 \end{cases}$$

$$q'(99.9) = 100(99.9 - 99.9) = 0$$
$$q'(100) = 100(100 - 99.9) = -100(100 - 100.1) = 10$$
$$q'(100.1) = -100(100.1 - 100.1) = 0$$

Hence, q' is continuous at the break points. Also, since $\hat{f}'(S) = 0$ for all S, $q'(S_1) = 0 = \hat{f}'(S_1)$ and $q'(S_3) = 0 = \hat{f}'(S_3)$. Hence, the end conditions are satisfied. Therefore, q is a quadratic interpolating spline for $\hat{f}$ at S_1, S_2, S_3. ■

(b) The graph:



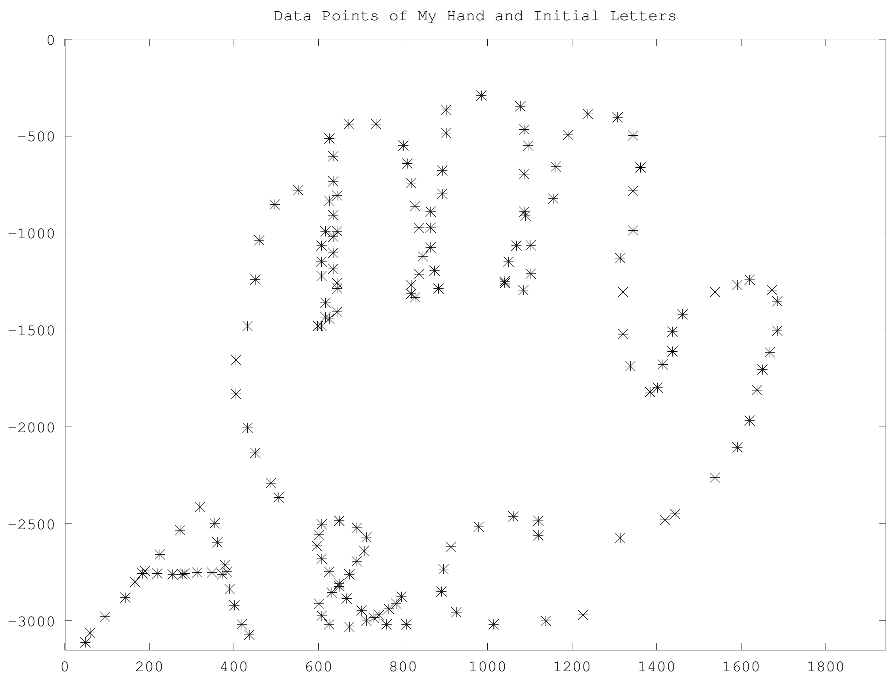
5. First, the end conditions of of not-a-knot cubic spline implies $S'''(x_1) = S'''(x_2)$ and $S'''(x_2) = S'''(x_3)$. Hence,

$$S'''(x_1) = S'''(x_2) = S'''(x_3)$$

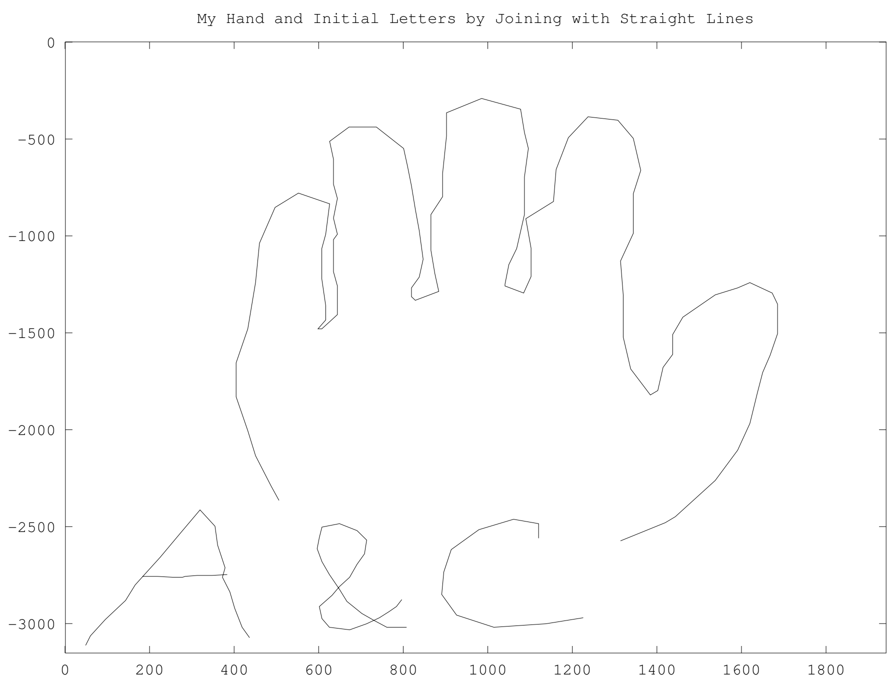
But $P'''(x) = 6$, a constant, which fits the end conditions of S. Secondly, P and its the first and second derivatives are continuous, because P is a polynomial (fact). Lastly, P maps all $(x_i, y_i), i \in \{1, 2, 3, 4\}$. Therefore, P is a not-a-knot spline interpolant for S. ■

6. The graphs:

(a) The Data Points

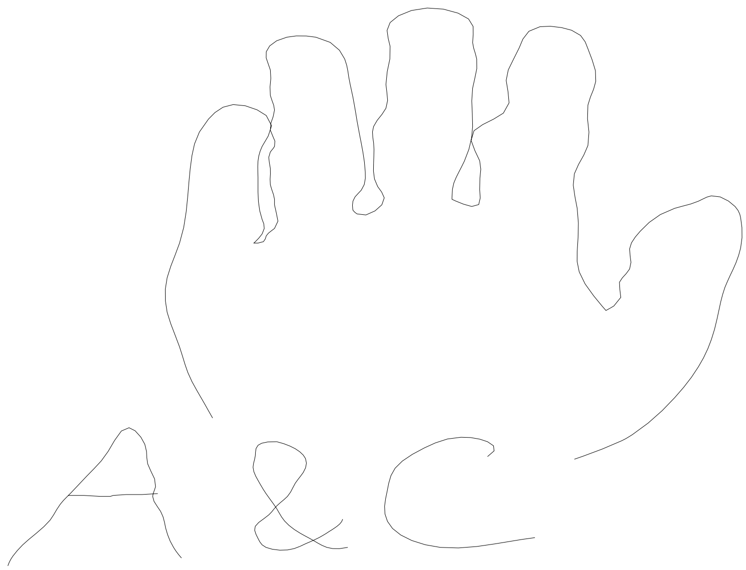


(b) Data Points Connected With Straight Lines



(c) Data Points With Interpolation

My Hand and Initial Letters by Interpolation



The codes (assuming that the splines package is already being loaded):

- init.m (only used in extraction of data points)

```
data = imread('hand_final.png');
clf;
h = imagesc(data);
axis image;
[x, y] = ginput;
```

- findt.m (helper function for the generation of the t parameters)

```
function [t] = findt(x,y)
  len = length(x);
  t = zeros(1, len);
  for p = 2:length(x)
    prev = p-1;
    distance = sqrt((x(prev) - x(p))^2 + (y(prev)-y(p))^2);
    t(p) = t(prev) + distance;
  end
endfunction
```

- refinet.m (helper function for refining t)

```
function [tref] = refinet(t,fac)
  N = length(t);
  tref = zeros(1, fac*(N-1) + 1);
```

```
  for k = 1:N-1
    i = fac*(k-1)+1;
    dt = t(k+1)-t(k);
    for l = 0:fac-1
      tref(i+l) = t(k) + l*dt/fac;
    end
  end
  tref(fac*(N-1)+1) = t(N);
endfunction
```

- interpolate.m (helper function that returns the interpolated vectors)

```
function [tx ty] = interpolate(x,y,end_type,refine_fac)
  t = findt(x,y);
  ppx = csape(t,x,end_type);
  ppy = csape(t,y,end_type);
  tref = refinet(t,refine_fac);
  tx = ppval(ppx,tref);
  ty = ppval(ppy,tref);
endfunction
```

- a1q6_data.m (the data)

```
% initals data
x0 = [48.833 60.161 95.407 143.239 165.897 189.813 225.058
   272.891 319.465 354.710 361.004 378.626 372.333 389.955
   401.284 418.907 436.529];
y0 = [3112.3 3063.2 2979.2 2881.0 2800.7 2743.1 2658.2 2533.5
    2413.6 2497.6 2595.8 2712.0 2761.1 2836.6 2920.7 3018.9
   3072.7];
x1 = [183.52 218.76 255.27 277.93 284.22 313.17 348.42
   383.66];
y1 = [2756.4 2756.4 2761.1 2761.1 2756.4 2751.6 2751.6
   2746.9];
x2 = [649.26 607.72 601.43 596.39 607.72 625.34 649.26 666.88
    702.13 731.08 761.29 807.86];
y2 = [2484.4 2502.3 2556.2 2613.8 2680.8 2746.9 2823.4 2885.7
    2948.0 2983.9 3018.9 3018.9];
x3 = [796.53 783.95 766.32 743.67 713.46 673.18 625.34 607.72
    601.43 631.64 649.26 673.18 690.80 708.42 713.46 690.80
   649.26];
y3 = [2876.3 2912.2 2938.6 2969.8 3000.9 3032.1 3018.9 2974.5
    2912.2 2854.6 2809.2 2761.1 2694.0 2640.2 2569.4 2520.3
   2484.4];
x4 = [1120.03 1120.03 1060.87 979.05 913.60 895.97 890.94
   926.18 1014.30 1137.66 1225.77];
```

```
y4 = [2559.9 2484.4 2461.7 2515.6 2618.5 2733.7 2849.8 2956.5
    3018.9 3000.9 2969.8];

% hand data
x5 = [506.07 487.64 450.77 432.34 404.69 404.69 432.34 450.77
    459.99 496.86 552.15 625.89 616.67 607.45 607.45 607.45
   616.67 616.67 598.24];
y5 = [2364.5 2290.8 2134.1 2005.1 1830 1654.8 1479.7 1240.1
   1037.3 853.01 779.28 834.58 991.26 1065 1147.9 1221.7
   1359.9 1433.6 1479.7];
x6 = [598.24 607.45 625.89 644.32 644.32 644.32 635.1 635.1
   635.1 644.32 635.1 644.32 635.1 635.1 625.89 671.97 736.48
   801 810.21 819.43 828.65 837.86 847.08 837.86 819.43
   819.43];
y6 = [1479.7 1479.7 1442.9 1406 1286.2 1258.5 1184.8 1101.9
   1018.9 991.26 908.31 806.93 733.2 604.17 512.01 438.28
   438.28 548.87 641.04 742.42 862.23 972.83 1120.3 1212.5
   1267.8 1313.8];
x7 = [819.43 828.65 883.94 874.73 865.51 865.51 865.51 893.16
    893.16 902.38 902.38 985.32 1077.5 1086.7 1095.9 1086.7
   1086.7 1068.3 1049.8 1040.6 1040.6];
y7 = [1313.8 1332.3 1286.2 1194 1074.2 972.83 889.88 797.72
   677.9 484.36 364.55 290.82 346.11 465.93 548.87 696.34
   889.88 1065 1147.9 1249.3 1258.5];
x8 = [1040.6 1084.8 1102.4 1102.4 1089.8 1155.3 1161.6 1190.5
    1237.1 1307.6 1344.1 1361.7 1344.1 1344.1 1313.9 1320.2
   1320.2 1337.8 1384.4];
y8 = [1258.5 1294.6 1209.7 1063.3 911.27 822.51 657.26 492.96
    385.31 403.26 496.74 661.98 781.9 986.81 1129.4 1304.1
   1522.2 1686.5 1820.6];
x9 = [1384.4 1402 1414.6 1437.2 1437.2 1461.2 1537.9 1590.8
   1619.8 1672.6 1685.2 1685.2 1667.6 1650 1637.4 1619.8
   1590.8 1537.9 1443.5 1419.6 1313.9];
y9 = [1820.6 1797.9 1678 1611 1509 1419.3 1304.1 1268.2
   1240.8 1294.6 1352.2 1504.3 1615.7 1704.4 1811.1 1967.9
   2105.8 2261.6 2448.5 2479.7 2573.2];
axis_dim = [1 1942 -3152 1];
```

- a1q6a.m (plot of the data points)

```
clear
a1q6_data; % load data

plot(x0,-y0,'*', x1,-y1,'*', x2,-y2,'*', x3,-y3,'*', x4,-y4,'
   *', x5, -y5,'*', x6,-y6,'*', x7,-y7,'*', x8,-y8,'*', x9,-y9
```

```
    ,'*');

title('Data Points of My Hand and Initial Letters');
axis(axis_dim);
```

- a1q6b.m (plot of the data points that are connected by lines)

```
clear
a1q6_data; % load data

plot(x0,-y0, x1,-y1, x2,-y2, x3,-y3, x4,-y4, x5,-y5, x6,-y6,
    x7,-y7, x8,-y8, x9,-y9);

title('My Hand and Initial Letters by Joining with Straight
    Lines');
axis(axis_dim);
```

- a1q6c.m (plot of the interpolated data)

```
clear
type = 'not-a-knot';
type2 = 'complete';
fac = 3;

a1q6_data; % load data

% interpolate initials
[tx0 ty0] = interpolate(x0,y0,type,fac);
[tx1 ty1] = interpolate(x1,y1,type,fac);
[tx2 ty2] = interpolate(x2,y2,type,fac);
[tx3 ty3] = interpolate(x3,y3,type,fac);
[tx4 ty4] = interpolate(x4,y4,type,fac);

% interpolate hand
[tx5 ty5] = interpolate(x5,y5,type2,fac);
[tx6 ty6] = interpolate(x6,y6,type2,fac);
[tx7 ty7] = interpolate(x7,y7,type2,fac);
[tx8 ty8] = interpolate(x8,y8,type2,fac);
[tx9 ty9] = interpolate(x9,y9,type2,fac);

plot(tx0,-ty0, x1,-y1, tx2,-ty2, tx3,-ty3, tx4,-ty4, tx5,-ty5
    , tx6,-ty6, tx7,-ty7, tx8,-ty8, tx9,-ty9);
title('My Hand and Initial Letters by Interpolation');
axis(axis_dim);
axis off;
```