Summer 2001 CS 222 - Homework 1 Solutions

1. (10 points) The standard formula and an alternative formula that also works are below:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{1}$$

$$x = \frac{-2c}{b \mp \sqrt{b^2 - 4ac}} \tag{2}$$

First, note that five-decimal digit arithmetic means base $\beta = 10$ and mantissa length $t = 5$. Therefore

$$\begin{aligned} a &= 1 = 0.10000 \times 10^1 \\ b &= 111.11 = .11111 \times 10^3 \Rightarrow b^2 = 0.12345 \times 10^4 \\ c &= 1.2121 = .12121 \times 10^1 \end{aligned}$$

$$\begin{aligned} \sqrt{b^2 - 4ac} &= \sqrt{0.12345 \times 10^5 - 0.48484 \times 10^1} \\ &= \sqrt{0.12345 \times 10^5 - 0.00005 \times 10^5} \\ &= \sqrt{0.12340 \times 10^5} \\ &= 0.11109 \times 10^2 \end{aligned}$$

(a) (3 points) Using formula (1),

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} = \frac{-.11111 \times 10^2 \pm .11109 \times 10^2}{.20000 \times 10^1} = -0.10000 \times 10^{-1} \quad \text{and} \quad -0.11110 \times 10^2$$

(b) (2 points) Using formula (2),

$$x = \frac{-2c}{b \mp \sqrt{b^2 - 4ac}} = \frac{-0.24242 \times 10^1}{0.11111 \times 10^2 \mp 0.11109 \times 10^2} = -.10910 \times 10^{-1} \quad \text{and} \quad -0.12121 \times 10^2$$

(c) (2 points) MATLAB code:

```
% problem 1 - part c
% solve for roots of the equation x^2 + 111.11x + 1.2121 = 0

clc
clear

% creates vector of coefficients
a=1; b=111.11; c=1.2121;
coeff=[a b c];

% finds and displays the roots
r=roots(coeff);
disp('Calculated roots to 5 significant digits are:')
disp(sprintf('%5.4e   ',r))
```

Output:

```
Calculated roots to 5 significant digits are:
-1.1110e+02   -1.0910e-02
```

(d) (3 points) Formula (1) gives a better estimate of the largest root (in absolute value), however formula (2) gives a better estimate of the smallest root (in absolute value). This is because b and $\sqrt{b^2-4ac}$ are similar in magnitude, and so $-b+\sqrt{b^2-4ac}$ loses significant digits (the result is 0.20000×10^{-4} which only has 1 significant digit). Using either formula, the root calculated using $-b+\sqrt{b^2-4ac}$ is the root that is incorrect. The incorrect results are due to cancellation error (discussed in class).

2. (10 points) MATLAB code:

```
% Compute (1+1/n)^n for n=10^k and k=1:20.  Compute the absolute and
% relative error between your estimates with the actual value of e,
% exp(1).
%
% Display a table with the values of n, approximated value of e,
% actual value of e, absolute error, and relative error.
clear
clc

n=logspace(1,20,20);
e=(1+(1./n)).^n;
exp1=exp(1);

% calculate absolute and relative error

abserr=abs(e-exp1);
relerr=abs((e-exp1)/exp1);

% prints table of values

disp('    n       e_approx   e_actual   abs.error   rel.error ')
disp('-------------------------------------------------------')
for i=1:20
  disp(sprintf('%3.2e    %5.4f     %5.4f     %5.4f     %5.4e',n(i),e(i),
       exp1,abserr(i),relerr(i)))
end
```

Output:

```
    n       e_approx   e_actual   abs.error   rel.error
-----------------------------------------------------------
1.00e+01     2.5937     2.7183     0.1245     4.5815e-02
1.00e+02     2.7048     2.7183     0.0135     4.9546e-03
1.00e+03     2.7169     2.7183     0.0014     4.9954e-04
1.00e+04     2.7181     2.7183     0.0001     4.9995e-05
1.00e+05     2.7183     2.7183     0.0000     4.9999e-06
1.00e+06     2.7183     2.7183     0.0000     5.0008e-07
1.00e+07     2.7183     2.7183     0.0000     4.9416e-08
1.00e+08     2.7183     2.7183     0.0000     1.1077e-08
1.00e+09     2.7183     2.7183     0.0000     8.2240e-08
1.00e+10     2.7183     2.7183     0.0000     8.2690e-08
1.00e+11     2.7183     2.7183     0.0000     8.2735e-08
1.00e+12     2.7185     2.7183     0.0002     8.8905e-05
1.00e+13     2.7161     2.7183     0.0022     7.9896e-04
1.00e+14     2.7161     2.7183     0.0022     7.9896e-04
1.00e+15     3.0350     2.7183     0.3168     1.1653e-01
1.00e+16     1.0000     2.7183     1.7183     6.3212e-01
1.00e+17     1.0000     2.7183     1.7183     6.3212e-01
1.00e+18     1.0000     2.7183     1.7183     6.3212e-01
1.00e+19     1.0000     2.7183     1.7183     6.3212e-01
1.00e+20     1.0000     2.7183     1.7183     6.3212e-01
```

Notice that the error decreases until $n = 12$ and then increases. This is due to rounding error. As n initially increases, we have better approximations of e, since ideally we need n large. However, after a certain point n is too large and $\frac{1}{n}$ rounds to 0 causing $1 + \frac{1}{n} = 1$. This explains the approximations of $e \approx 1$ when n is largest.

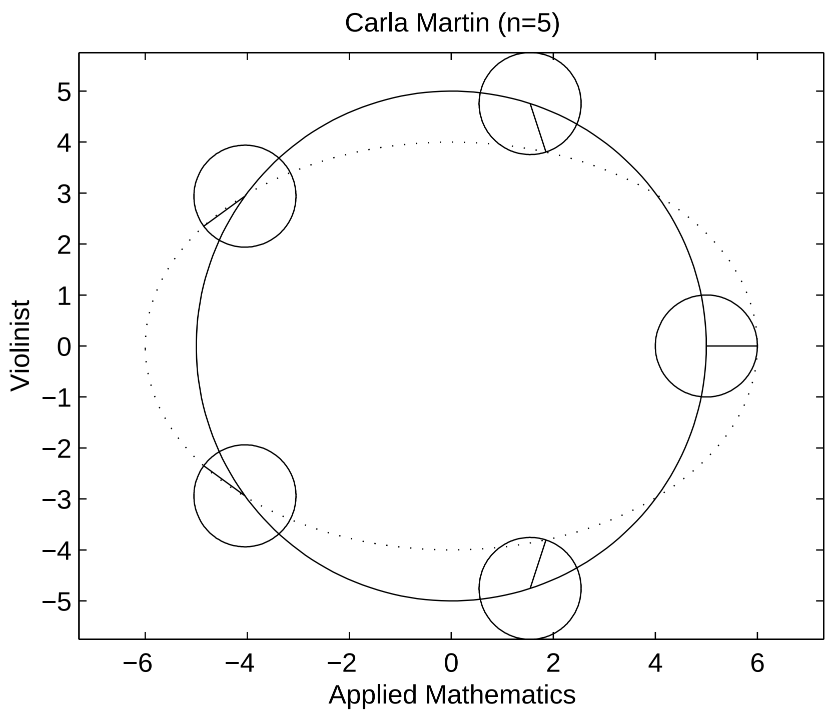
3. (20 points) MATLAB code:

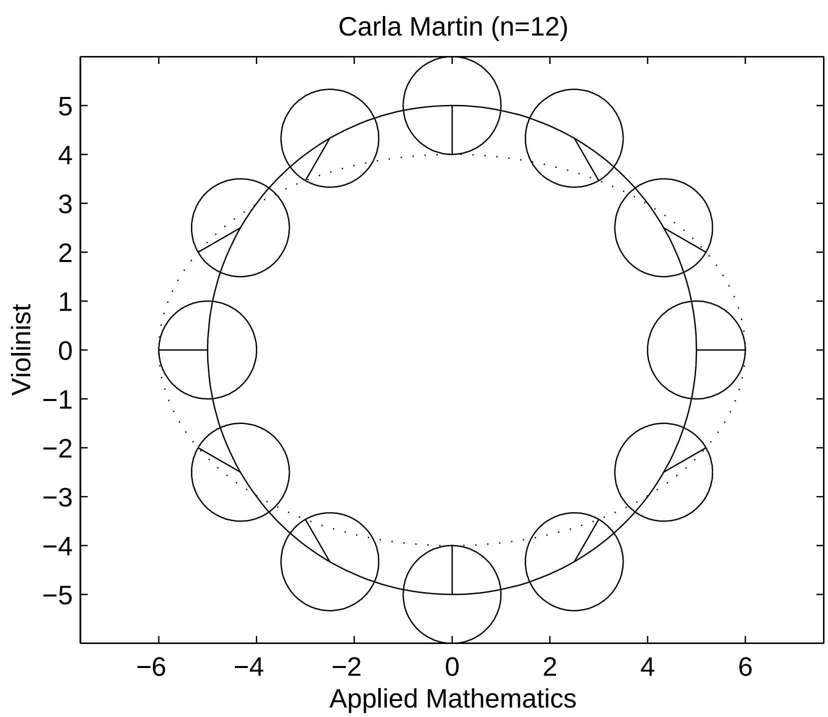
```
% Let a and b be real numbers with b<a.  Let n be a positive
% integer.  Let r1=(a+b)/2 and r2=(a-b)/2.
% In the same figure, draw the ellipse with a dotted line
%
%       (a*cos(t),b*sin(t))       0<=t<=2*pi,
%
% the ``big'' circle with a solid line
%
%       (r1*cos(t),r1*sin(t))      0<=t<=2*pi,
%
% and n ``small'' circles with solid lines.
% The kth small circle should have radius
% r2 and center (r1*cos(2*pi*k/n),r1*sin(2*pi*k/n)).  A radius making
% angle -2*pi*k/n should be drawn inside the kth small circle.

close all
clear

% assigning initial variables
```

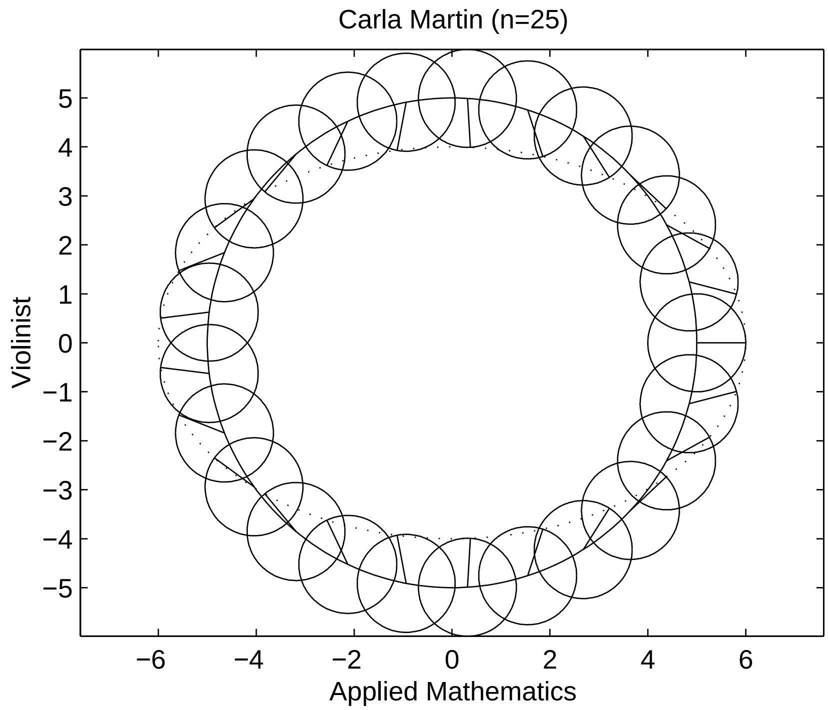
```
a=6;
b=4;
t=linspace(0,2*pi,200);
cost=cos(t);   % assigning so only have to calculate this once
sint=sin(t);

n=[5 12 25];

r1=(a+b)/2;
r2=(a-b)/2;

for j=1:length(n)
  figure

  % plots the ellipse and big circle
  plot(a*cost,b*sint,':',r1*cost,r1*sint,'-')

  hold on

  % small circle calculations
  for k=1:n(j)

    % creating variables for calculations used more than once
    angle=2*pi*k/n(j);
    cosangle=cos(angle);
    sinangle=sin(angle);
    centerx=r1*cosangle;
    centery=r1*sinangle;

    % plots n small circles
    plot((r2*cost+centerx),(r2*sint+centery));

    % to plot the radii of the small circles, we know the radius and the angle
    % (polar coordinates) and so to convert to (x,y) coordinates, use
    % x=r cos(theta),y=r sin(theta).  Since the center of each circle changes,
    % adjust the (x,y) points by the center.

    % this involves cos(-angle)=cos(angle) and sin(-angle)=-sin(angle)

    x=r2*cosangle + centerx;
    y=r2*(-sinangle) + centery;

    plot([centerx x],[centery,y]);
  end

  % graph specifications
  axis equal
  title(sprintf('Carla Martin (n=%d)',n(j)))
  xlabel('Applied Mathematics')
  ylabel('Violinist')
```

```
  hold off
end
```

Output:






Carla Martin (n=25)
Violinist
5
4
3
2
1
0
−1
−2
−3
−4
−5
−6
−4
−2
0
2
4
6
Applied Mathematics