Project A - elliptic integrals

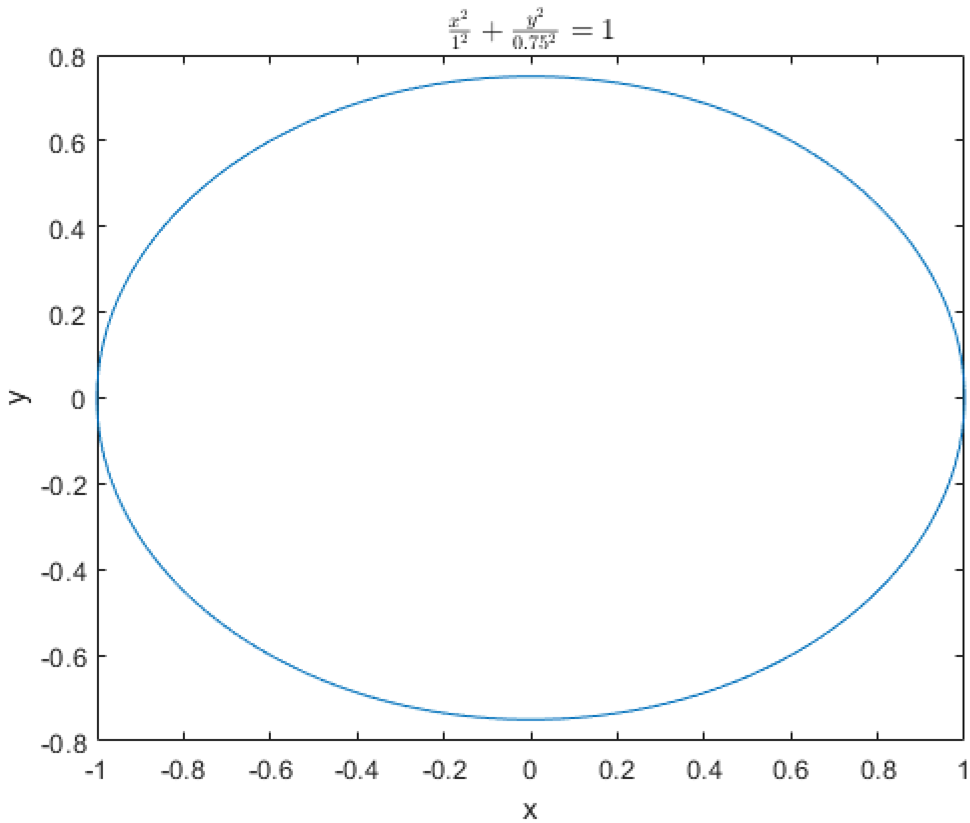
Table of Contents

Exercise A1 1
Exercise A2 2
Exercise A3 3

Candidate number 1045139

Exercise A1

```
type ellipse.m
%Get points for (a,b,n) = (1,0.75,800)
[x,y] = ellipse(1,0.75,800);
%Plot the ellipse x^2 + y^2/.75^2 = 1
plot(x,y)
%Add axis titles, etc.
xlabel("x")
ylabel("y")
title("$\frac{x^2}{1^2} + \frac{y^2}{0.75^2} =
 1$","interpreter","latex")


function [x,y] = ellipse(a,b,n)
%Exercise A1
    %Generate equally spaced points along the ellipse
    t = linspace(0*2*pi/n,2*pi,n);
    x = a.*cos(t);
    y = b.*sin(t);
end
```



Exercise A2

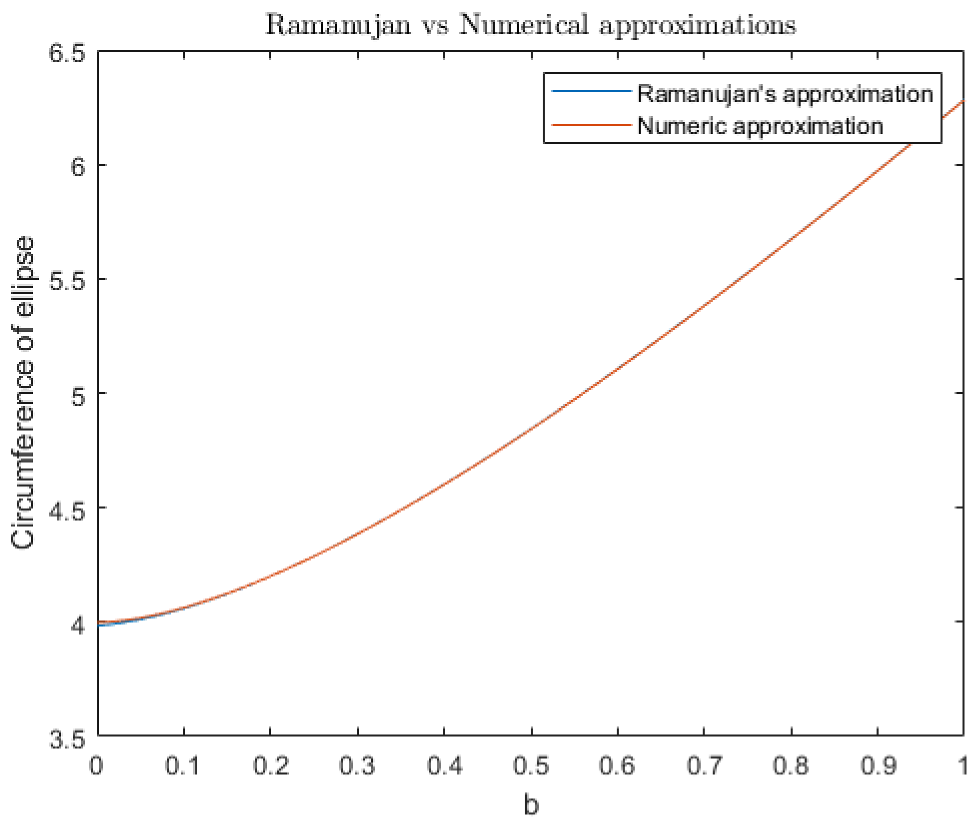
```
type arclength.m
%Check arclength of a circle of radius a by iterating over values of a
for i = 1:5
    %Get points of ellipse
    [x,y] = ellipse(i/5,i/5,100);
    %Calculate & display arclength
    l = arclength(x,y)
    %Calculate difference from true value
    disp("Difference from true value =" + (l - 2*pi*i/5))
end


function [len] = arclength(x,y)
%Exercise A2
    %Get n, the number of elements of each of the input sets
    n = length(x);
    %Initialize the length
    len = 0;
    %Iterate from 1 to n-1
    for i = 1:(n-1)
        %Calculate the length of the sides of each sector
        xdiff = x(i+1) - x(i);
        ydiff = y(i+1) - y(i);
        %Add the hypotenuse of each sector to len
```

```
        len = len + sqrt(xdiff.^2 + ydiff.^2);
    end
end

l =

    1.2564

Difference from true value =-0.00021089

l =

    2.5129

Difference from true value =-0.00042179

l =

    3.7693

Difference from true value =-0.00063268

l =

    5.0257

Difference from true value =-0.00084358

l =

    6.2821

Difference from true value =-0.0010545

%The arclength function consistently underestimates the true
 arclength. This is because an ellipse is concave outward, so the
 line segments are each slightly shorter than the arc segments they
 represent.
```

Exercise A3

```
%Define a function for Ramanujan's approximation.
C_ram = @(a,b) pi.*(3.*(a+b)-sqrt((3.*a+b).*(a+3.*b)));
%Generate the list of values of b
b = linspace(0,1,100);
%Plot Ramanujan's approximation against b
plot(b,C_ram(1,b));
%Don't clear the plot
hold on
%Create an empty list for numeric approximations
app = zeros(100);
%Calculate numeric approximations for each value of b
for i = 1:length(b)
```

```
    [x,y] = ellipse(1,b(i),100);
    app(i) = arclength(x,y);
end
%Plot numeric approximations against b
plot(b,app);
%Add legend, axis titles, etc.
legend("Ramanujan's approximation","Numeric approximation")
xlabel("b")
ylabel("Circumference of ellipse")
ylim([3.5,6.5])
title("Ramanujan vs Numerical approximations","interpreter","latex")
```




```
%Both approximations are very close to each other, barely
 distinguishable. They diverge as b shrinks, with the biggest
 difference where b=0. Then Ramanujan's approximation gives a value of
 3.983 while numerical approximation gives a value of 3.999.
```

Published with MATLAB® R2019b