
Lab7Simons.m

Table of Contents

Problem 1	1
Problem 2	1
Problem 3	2

Madilyn Simons

Problem 1

Approximate the integral of a function once using the Composite Trapezoidal rule and once using the Composite Simpson's rule.

```
% function to approximate the integral of
f = @(x) 1 ./ (x .* log(x));

% lowest end-point of integral
a = exp(1);

% upper end-point of integral
b = exp(1) + 2;

% number of subintervals to use
n = 8;

% approximate using Composite Trapezoidal Rule
t_approx = composite_trapezoidal_rule(f, a, b, n);

% approximate using Composite Simpson's rule
s_approx = composite_simpsons_rule(f, a, b, n);

% Print the results
fprintf('Approximation using Composite Trapezoidal Rule\n');
fprintf('%f\n\n', t_approx);

fprintf('Approximation using Composite Simpsons Rule\n');
fprintf('%f\n\n', s_approx);

Approximation using Composite Trapezoidal Rule
0.440345

Approximation using Composite Simpsons Rule
0.439199
```

Problem 2

Use the Composite Trapezoidal rule to approximate the area averaged

```
lining temperature of a brake pad

% a series of radii on the pad-disk
r = [0.308 0.325 0.342 0.359 0.376 0.393 0.410 0.427 0.444 0.461
0.478];

% the temperature of the pad disk at radius r(i)
t = [640 794 885 943 1034 1064 1114 1152 1204 1222 1239];

% the angle subtended by the sector brake pads
theta_p = 0.7051;

% find the sum of the inner nodes of T(r)*r*theta
sum = 0;
for i=2:10
    sum = sum + t(i) * r(i) * theta_p;
end

% the sum of the outter nodes of T(r)*r*theta
k = theta_p * (t(1) * r(1) + t(11) * r(11));

% an approximate value of the integral of T(r)*r*theta from re to ro
top = (0.0170 / 2.0) * (2 * sum + k);

% the sum of the inner nodes of r*theta
sum = 0;
for i=2:10
    sum = sum + r(i) * theta_p;
end

% the sum of the outter nodes of r*theta
k = theta_p * (r(1) + r(11));

% an approximate value of the integral of r*theta from re to ro
bottom = (0.0170 / 2.0) * (2 * sum + k);

% approximate value of T
T = top / bottom;

% print the result
fprintf('T = %.3f\n', T);

T = 1055.327
```

Problem 3

```
Approximate the length of the Gateway Arch using the Composite Simpson's
rule

a = -91.20;
b = 91.20;

% derivative of the equation used to constuct the Gateway Arch
dy = @(x) 0.6899529 .* sin(0.03291765 .* x);
```

```
% derivative of arc length
ds = @(x) sqrt(1 + (dy(x)).^2);

% find how many nodes are needed to approx the length to the nearest
meter
n = how_many_nodes(ds, a, b, 1);

% approximate the arc length by approximating the integral of ds
% and round it to the nearest meter
arc_length = composite_simpsons_rule(ds, a, b, n);
arc_length = round(arc_length);

% print the results
fprintf('Arc length = %d meters\n', arc_length);

Arc length = 203 meters
```

Published with MATLAB® R2018b