

UWTSD

MSc

Task 1

Module code: ACHA7002

Student number: 2200899

7/12/2025

Contents

1	Plotting and differentiation.....	1
2	Trapezoidal Method.....	2
3	Composite	2
4	Simpsons.....	2
5	Integration results tables.....	4
5.1	Composite convergence iterations	4
5.2	Simpsons Convergence iterations.....	11
6	Appendices.....	11
6.1	Your code script and functions.....	11

1 Plotting and differentiation

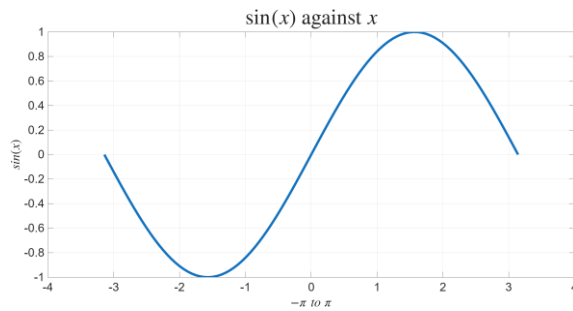


Figure 1 $y = \sin(x)$ plot

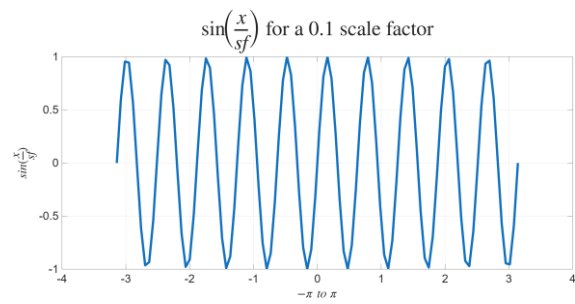


Figure 2 $y = \sin_scale(x, sf)$ plot (0.1 steps)

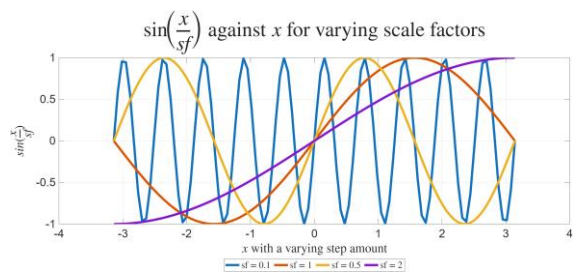


Figure 3 $y = \sin_scale(x, sf)$ plot ([0.1 0.5 2] steps)

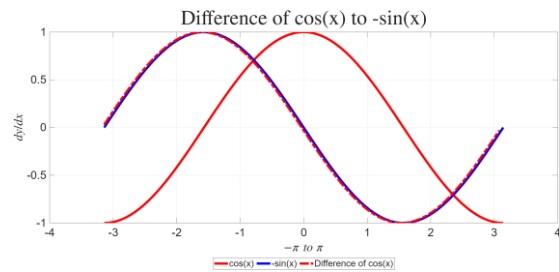


Figure 4 diff $\cos(x)$ to $-\sin(x)$

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{h_i}$$

Equation 1-1 numerical differentiation

2 Trapezoidal Method

$$I \approx \frac{1}{2}(f_a + f_b)(b - a)$$

Equation 2-1 trapezoidal method

A function was created to integrate 'x sin(x)' between 0 and pi/2 the code is in appendix

$$I = \int_0^{\pi/2} x \sin(x) dx ;$$

IntTrap (your code result)	Analytical (result)	percentage_error_Trap
1.234	1	18.963%

3 Composite

A function was created to integrate further functions in a discretised way from equation 3.1.

$$I \approx \frac{1}{2} \Delta x_n (f_a + 2 \sum_{i=1}^{n-1} f_i + f_b)$$

Equation 3-1 composite

Table 3-1 results for integration of $f(x) = 1/x$ using composite method a while loop was used to find when the result was within $1e-6$

	Analytical	Composite Trapezium			Converge $1e-6$
'Result'	0.6931472	0.6965349	0.6931722	0.6931534	0.6931479
'n iterations'	Na	5	50	100	301
'percentage difference'	Na	0.4888	3.6068×10^{-3}	8.9447×10^{-4}	1.0099×10^{-4}

4 Simpsons

A function was created to integrate further functions in a discretised separating odd and even crossing over in a quadratic manner way from equation 4.1

$$I \approx \frac{1}{3} \Delta x_n (f_a + 4 \sum_{i=1, \text{ odd only}}^{n-1} f_i + 2 \sum_{i=1, \text{ even only}}^{n-1} f_i + f_b)$$

Equation 4-1 Simpsons

Table 4-1 results for integration of $f(x) = 1/x$ using Simpson's method a while loop was used to find when the result was within $1e-6$ of the analytical

	Analytical	Simpsons Trapezium			Converge 1e-6
'Result'	0.69314718	0.58783069	0.69314719	0.69314718	0.69314765
'n iterations'	NA	5	50	100	18
'percentage difference'	NA	15.194	1.443x10-6	0	6.781x10-5

5 Integration results tables

5.1 Composite convergence iterations

Iteration	Convergence	Value
1	0.082	0.75
2	0.0219	0.7083
3	0.0099	0.7
4	0.0056	0.697
5	0.0036	0.6956
6	0.0025	0.6949
7	0.0018	0.6944
8	0.0014	0.6941
9	0.0011	0.6939
10	9.01E-04	0.6938
11	7.44E-04	0.6937
12	6.26E-04	0.6936
13	5.33E-04	0.6935
14	4.60E-04	0.6935
15	4.01E-04	0.6934
16	3.52E-04	0.6934
17	3.12E-04	0.6934
18	2.78E-04	0.6933
19	2.50E-04	0.6933
20	2.25E-04	0.6933
21	2.04E-04	0.6933
22	1.86E-04	0.6933
23	1.70E-04	0.6933
24	1.57E-04	0.6933
25	1.44E-04	0.6932
26	1.33E-04	0.6932
27	1.24E-04	0.6932
28	1.15E-04	0.6932
29	1.07E-04	0.6932
30	1.00E-04	0.6932
31	9.38E-05	0.6932
32	8.80E-05	0.6932
33	8.28E-05	0.6932
34	7.80E-05	0.6932
35	7.36E-05	0.6932
36	6.96E-05	0.6932
37	6.59E-05	0.6932
38	6.24E-05	0.6932
39	5.93E-05	0.6932
40	5.64E-05	0.6932

41	5.36E-05	0.6932
42	5.11E-05	0.6932
43	4.88E-05	0.6932
44	4.66E-05	0.6932
45	4.45E-05	0.6932
46	4.26E-05	0.6932
47	4.08E-05	0.6932
48	3.91E-05	0.6932
49	3.76E-05	0.6932
50	3.61E-05	0.6932
51	3.47E-05	0.6932
52	3.33E-05	0.6932
53	3.21E-05	0.6932
54	3.09E-05	0.6932
55	2.98E-05	0.6932
56	2.88E-05	0.6932
57	2.78E-05	0.6932
58	2.68E-05	0.6932
59	2.59E-05	0.6932
60	2.50E-05	0.6932
61	2.42E-05	0.6932
62	2.35E-05	0.6932
63	2.27E-05	0.6932
64	2.20E-05	0.6932
65	2.13E-05	0.6932
66	2.07E-05	0.6932
67	2.01E-05	0.6932
68	1.95E-05	0.6932
69	1.89E-05	0.6932
70	1.84E-05	0.6932
71	1.79E-05	0.6932
72	1.74E-05	0.6932
73	1.69E-05	0.6932
74	1.65E-05	0.6932
75	1.60E-05	0.6932
76	1.56E-05	0.6932
77	1.52E-05	0.6932
78	1.48E-05	0.6932
79	1.44E-05	0.6932
80	1.41E-05	0.6932
81	1.37E-05	0.6932
82	1.34E-05	0.6932
83	1.31E-05	0.6932
84	1.28E-05	0.6932

85	1.25E-05	0.6932
86	1.22E-05	0.6932
87	1.19E-05	0.6932
88	1.16E-05	0.6932
89	1.14E-05	0.6932
90	1.11E-05	0.6932
91	1.09E-05	0.6932
92	1.07E-05	0.6932
93	1.04E-05	0.6932
94	1.02E-05	0.6932
95	9.99E-06	0.6932
96	9.78E-06	0.6932
97	9.58E-06	0.6932
98	9.39E-06	0.6932
99	9.20E-06	0.6932
100	9.02E-06	0.6932
101	8.84E-06	0.6932
102	8.67E-06	0.6932
103	8.50E-06	0.6932
104	8.34E-06	0.6932
105	8.18E-06	0.6932
106	8.02E-06	0.6932
107	7.88E-06	0.6932
108	7.73E-06	0.6932
109	7.59E-06	0.6932
110	7.45E-06	0.6932
111	7.32E-06	0.6932
112	7.19E-06	0.6932
113	7.06E-06	0.6932
114	6.94E-06	0.6932
115	6.82E-06	0.6932
116	6.70E-06	0.6932
117	6.59E-06	0.6932
118	6.48E-06	0.6932
119	6.37E-06	0.6932
120	6.26E-06	0.6932
121	6.16E-06	0.6932
122	6.06E-06	0.6932
123	5.96E-06	0.6932
124	5.86E-06	0.6932
125	5.77E-06	0.6932
126	5.68E-06	0.6932
127	5.59E-06	0.6932
128	5.50E-06	0.6932

129	5.42E-06	0.6932
130	5.34E-06	0.6932
131	5.25E-06	0.6932
132	5.17E-06	0.6932
133	5.10E-06	0.6932
134	5.02E-06	0.6932
135	4.95E-06	0.6932
136	4.88E-06	0.6932
137	4.80E-06	0.6932
138	4.73E-06	0.6932
139	4.67E-06	0.6932
140	4.60E-06	0.6932
141	4.54E-06	0.6932
142	4.47E-06	0.6932
143	4.41E-06	0.6932
144	4.35E-06	0.6932
145	4.29E-06	0.6932
146	4.23E-06	0.6932
147	4.17E-06	0.6932
148	4.12E-06	0.6932
149	4.06E-06	0.6931
150	4.01E-06	0.6931
151	3.95E-06	0.6931
152	3.90E-06	0.6931
153	3.85E-06	0.6931
154	3.80E-06	0.6931
155	3.75E-06	0.6931
156	3.71E-06	0.6931
157	3.66E-06	0.6931
158	3.61E-06	0.6931
159	3.57E-06	0.6931
160	3.52E-06	0.6931
161	3.48E-06	0.6931
162	3.44E-06	0.6931
163	3.39E-06	0.6931
164	3.35E-06	0.6931
165	3.31E-06	0.6931
166	3.27E-06	0.6931
167	3.23E-06	0.6931
168	3.19E-06	0.6931
169	3.16E-06	0.6931
170	3.12E-06	0.6931
171	3.08E-06	0.6931
172	3.05E-06	0.6931

173	3.01E-06	0.6931
174	2.98E-06	0.6931
175	2.94E-06	0.6931
176	2.91E-06	0.6931
177	2.88E-06	0.6931
178	2.85E-06	0.6931
179	2.81E-06	0.6931
180	2.78E-06	0.6931
181	2.75E-06	0.6931
182	2.72E-06	0.6931
183	2.69E-06	0.6931
184	2.66E-06	0.6931
185	2.63E-06	0.6931
186	2.61E-06	0.6931
187	2.58E-06	0.6931
188	2.55E-06	0.6931
189	2.52E-06	0.6931
190	2.50E-06	0.6931
191	2.47E-06	0.6931
192	2.45E-06	0.6931
193	2.42E-06	0.6931
194	2.40E-06	0.6931
195	2.37E-06	0.6931
196	2.35E-06	0.6931
197	2.32E-06	0.6931
198	2.30E-06	0.6931
199	2.28E-06	0.6931
200	2.25E-06	0.6931
201	2.23E-06	0.6931
202	2.21E-06	0.6931
203	2.19E-06	0.6931
204	2.17E-06	0.6931
205	2.15E-06	0.6931
206	2.12E-06	0.6931
207	2.10E-06	0.6931
208	2.08E-06	0.6931
209	2.06E-06	0.6931
210	2.04E-06	0.6931
211	2.03E-06	0.6931
212	2.01E-06	0.6931
213	1.99E-06	0.6931
214	1.97E-06	0.6931
215	1.95E-06	0.6931
216	1.93E-06	0.6931

217	1.91E-06	0.6931
218	1.90E-06	0.6931
219	1.88E-06	0.6931
220	1.86E-06	0.6931
221	1.85E-06	0.6931
222	1.83E-06	0.6931
223	1.81E-06	0.6931
224	1.80E-06	0.6931
225	1.78E-06	0.6931
226	1.77E-06	0.6931
227	1.75E-06	0.6931
228	1.73E-06	0.6931
229	1.72E-06	0.6931
230	1.70E-06	0.6931
231	1.69E-06	0.6931
232	1.68E-06	0.6931
233	1.66E-06	0.6931
234	1.65E-06	0.6931
235	1.63E-06	0.6931
236	1.62E-06	0.6931
237	1.61E-06	0.6931
238	1.59E-06	0.6931
239	1.58E-06	0.6931
240	1.57E-06	0.6931
241	1.55E-06	0.6931
242	1.54E-06	0.6931
243	1.53E-06	0.6931
244	1.51E-06	0.6931
245	1.50E-06	0.6931
246	1.49E-06	0.6931
247	1.48E-06	0.6931
248	1.47E-06	0.6931
249	1.45E-06	0.6931
250	1.44E-06	0.6931
251	1.43E-06	0.6931
252	1.42E-06	0.6931
253	1.41E-06	0.6931
254	1.40E-06	0.6931
255	1.39E-06	0.6931
256	1.38E-06	0.6931
257	1.37E-06	0.6931
258	1.35E-06	0.6931
259	1.34E-06	0.6931
260	1.33E-06	0.6931

261	1.32E-06	0.6931
262	1.31E-06	0.6931
263	1.30E-06	0.6931
264	1.29E-06	0.6931
265	1.28E-06	0.6931
266	1.27E-06	0.6931
267	1.26E-06	0.6931
268	1.26E-06	0.6931
269	1.25E-06	0.6931
270	1.24E-06	0.6931
271	1.23E-06	0.6931
272	1.22E-06	0.6931
273	1.21E-06	0.6931
274	1.20E-06	0.6931
275	1.19E-06	0.6931
276	1.18E-06	0.6931
277	1.18E-06	0.6931
278	1.17E-06	0.6931
279	1.16E-06	0.6931
280	1.15E-06	0.6931
281	1.14E-06	0.6931
282	1.13E-06	0.6931
283	1.13E-06	0.6931
284	1.12E-06	0.6931
285	1.11E-06	0.6931
286	1.10E-06	0.6931
287	1.09E-06	0.6931
288	1.09E-06	0.6931
289	1.08E-06	0.6931
290	1.07E-06	0.6931
291	1.06E-06	0.6931
292	1.06E-06	0.6931
293	1.05E-06	0.6931
294	1.04E-06	0.6931
295	1.04E-06	0.6931
296	1.03E-06	0.6931
297	1.02E-06	0.6931
298	1.02E-06	0.6931
299	1.01E-06	0.6931
300	1.00E-06	0.6931
301	9.95E-07	0.6931

5.2 Simpsons Convergence iterations

Iteration	Convergence	Value
2	0	0
4	0.0019	0.6944
6	1.54E-04	0.6933
8	3.26E-05	0.6932
10	1.06E-05	0.6932
12	4.40E-06	0.6932
14	2.14E-06	0.6931
16	1.16E-06	0.6931
18	6.81E-07	0.6931

6 Appendices

6.1 Your code script and functions

```

clc, clear, close all

format short g

N=100;
x = linspace(-pi,pi,N);

y = sin(x);

figure;
plot(x,y,LineWidth=5)
xlabel('$-\pi$ to $\pi$', 'Interpreter', 'latex', FontSize=30)
ylabel('$\sin(x)$', 'Interpreter', 'latex', FontSize=30)
title('$\sin(x)$ against $x$', 'Interpreter', 'latex', FontSize=45)
grid on
ax = gca;
ax.XAxis.FontSize = 25; % x-axis numbers
ax.YAxis.FontSize = 25; % y-axis numbers

```

Figure 5 - $\sin(x)$ main script code

```

function [y] = sin_scale(x,sf)

y = sin(x./sf);

end

```

Figure 6 - $\sin_Scale$ function used in part 1

```

sf = 0.1;
y = sin_scale(x,sf);

figure;
plot(x,y,LineWidth=5)
xlabel('$-\pi$ to $\pi$', 'Interpreter', 'latex', FontSize=30)
ylabel('$\sin(\frac{x}{sf})$', 'Interpreter', 'latex', FontSize=30)
title('$\sin(\frac{x}{sf})$ for a 0.1 scale factor', 'Interpreter', 'latex', FontSize=45)
grid on
ax = gca;
ax.XAxis.FontSize = 25; % x-axis numbers
ax.YAxis.FontSize = 25; % y-axis numbers

```

Figure 7 - $\sin_scale(x, sf)$ plot (0.1 step) main script code

```

sf=[0.1 1 0.5 2]; % Scale factor values in an array named sf
N_sf = numel(sf); % Number of elements within sf (ML intrinsic % function)
sin_array = zeros(N,N_sf);
%Initialise an array of zeros to store data. % Notice the size of the array is predicted by
%the number of values, N and the number of scale factors to be analysed, N_sf.

for ndx = 1: N_sf
    sin_array(:,ndx) = sin_scale(x,sf(ndx));
end
figure;
plot(x,sin_array,LineWidth=5)
labels = compose('sf = %g', sf); % automatically make label strings
legend(labels, 'Location', 'best', 'Interpreter', 'latex','location','southoutside','Orientation','horizontal',FontSize=20)
xlabel('$x$ with a varying step amount', 'Interpreter', 'latex', FontSize=30)
ylabel('$\sin(\frac{x}{sf})$', 'Interpreter', 'latex', FontSize=30)
title('$\sin(\frac{x}{sf})$ against $x$ for varying scale factors', ...
      'Interpreter', 'latex', FontSize=45)
grid on
ax = gca;
ax.XAxis.FontSize = 25; % x-axis numbers
ax.YAxis.FontSize = 25; % y-axis numbers

```

Figure 8 - $\sin_scale(x, sf)$ (1,0.1,0.5,2 step) main script code

```

function [dy_dx] = forward_diff_tech(x,y)

h = x(2) - x(1);
dy_dx= (y(2:end)-y(1:end-1))/h;

end

```

Figure 9 - Forward different technique function used in part 1

```

N=100;
x = linspace(-pi,pi,N);

y=cos(x);

dydx = forward_diff_tech(x, y);

figure;
plot(x, cos(x), 'r',LineWidth=5) % analytical derivative
hold on
plot(x, -sin(x), 'b',LineWidth=5)
plot(x(1:end-1), dydx, 'r.-.',LineWidth=5) % numerical derivative
xlabel('$-\pi$ to $\pi$', 'Interpreter', 'latex', FontSize=30)
ylabel('$dy/dx$', 'Interpreter', 'latex', FontSize=30)
legend('cos(x)', '-sin(x)', 'Difference of cos(x)', 'location', 'southoutside', 'Orientation', 'horizontal', FontSize=20)
title('Difference of cos(x) to -sin(x)', 'Interpreter', 'latex', FontSize=45)
grid on
ax = gca;
ax.XAxis.FontSize = 25; % x-axis numbers
ax.YAxis.FontSize = 25; % y-axis numbers
hold off

```

Figure 10 - $\text{diff} \cos(x)$ to $-\sin(x)$ main script code

```

function int=trap1(f,a,b)

if a>b
error('b must be greater than a')
end
h=(b-a);
int=h.*(feval(f,a)+feval(f,b))/2;
end

```

Figure 11 - Trapezium method function used in part 2

```

N = 100;% doesnt matter its just used to plot the x
x = linspace(-pi, pi, N);

% Get actual x values for a and b
a = 0;
b = pi/2;

% Define the function
f = @(x) x .* sin(x);

int = trap1(f, a, b);          % compute integral between a and b

% --- PLOTTING ---

figure;
plot(x,f(x))
hold on

fa = f(a);
fb = f(b);
plot([a b],[fa fb])

plot([a a],[min(f(x)) fa],'r--');
plot([b b],[min(f(x)) fb],'r--');
xlabel('$-\pi$ to $\pi$', 'Interpreter', 'latex',FontSize=30)
ylabel('$x\sin(x)$', 'Interpreter', 'latex',FontSize=30);
title('Trapezium method','Interpreter','latex',FontSize=45)
legend('True function', 'Trapezium rule','location','southoutside','Orientation','horizontal',FontSize=20)
grid on
ax = gca;
ax.XAxis.FontSize = 25;    % x-axis numbers
ax.YAxis.FontSize = 25;    % y-axis numbers
hold off

fprintf('Trapezium method %f\n', sum(int))

clear
syms x
expr = x * sin(x);
True_value = int(expr, x, 0, (pi/2));
fprintf('True_value = %f', double(True_value));

```

Figure 12 – Trapezium method main script ($x\sin(x)$)

```
function int_total = trap2(f, a1, b1, N)

if a1 >= b1
    error('Upper limit (b) must be greater than lower limit (a)');
end

% Step size
h = (b1 - a1) / N;

% Subinterval points
x = linspace(a1, b1, N + 1);

% Function values at each point
fx = f(x);

% Composite trapezium rule formula
int_total = (h/2) * (fx(1) + 2*sum(fx(2:end-1)) + fx(end));

end
```

Figure 13 - Composite trapezium rule function used in part 3

```

% Part 5 - Composite trapezium rule

f = @(x) (1./x);      % Integrand
Comp_N = [5, 50, 100]; % Different subinterval counts
a1 = 1;
b1 = 2;
N_old = 100; % doesn't matter its just used to plot the x
x_old = linspace(-pi, pi, N_old);

for i = 1:length(Comp_N)
    N = Comp_N(i); % number of trapeziums
    x = linspace(a1, b1, N+1); % N subintervals = N+1 points
    h = (b1 - a1) / N; % step size

    % Composite Trapezium Rule formula
    fx = f(x);
    % int_total = (h/2) * (fx(1) + 2*sum(fx(2:end-1)) + fx(end));
    int_total = trapz(f, a1, b1, N); % compute integral between a and b

    % ---- Plotting section ----
    figure;
    plot(x_old, f(x_old), 'k', 'linewidth', 1.5) % True curve
    hold on

    % Draw each trapezium segment
    for i = 1:N
        a = x(i);
        b = x(i+1);
        fa = f(a);
        fb = f(b);

        % Draw trapezium sides
        plot([a b], [fa fb], 'b')

        % Draw vertical dashed red lines
        plot([a a], [0 fa], 'r--')
        plot([b b], [0 fb], 'r--')
    end

    xlabel('$x$', 'Interpreter', 'latex')
    ylabel('$f\left(\frac{1}{x}\right)$', 'Interpreter', 'latex');
    title(['Composite Trapezium Rule using ', num2str(N), ' subintervals'])
    legend('True function', 'Trapezium edges')
    grid on

    % ---- Output integral value ----
    fprintf('Composite Trapezium (N=%d): %.7f\n', N, int_total);
end

% ---- Compare to true symbolic value ----
syms x
True_value = int(1./x, x, a1, b1);
fprintf('True integral value = %.7f\n', double(True_value));

```

Figure 14 - Composite trapezium rule main script code