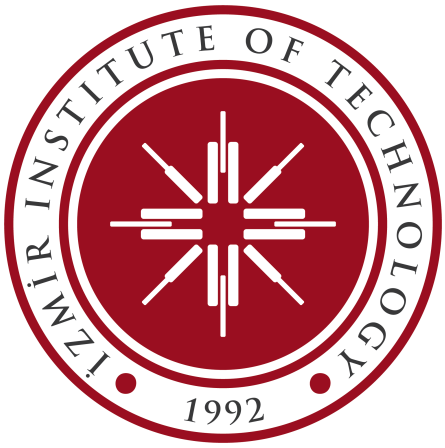

İzmir Institute of Technology

Numerical Methods in Engineering

CE301

Assignment #1

Section B

Mert Emrem - 250203015

Mechanical Engineering

Summer 2021

To compare for error, the value of the integral that is accurate to more decimal places than are required is obtained via WolframAlpha:

$$\int_{\pi/6}^{\pi/2} \frac{\cos(x)(e^x + x)}{x^2 - \ln(x)} \, \mathrm{d}x = 1.6724695$$

The MATLAB code (see Appendix A.1) output is as follows:

```
The number of sub-intervals needed for:

----> Compound mid-point: 172
----> Compound trapezoid: 243
----> Simpson's rule:     16
```

The code can be further deliberated upon at instructors' request.

Appendix

A.1 Computer Code

```
%%%%%%%%%%%%%%%%%%%%%%    Mert Emrem - 250203015    %%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%    CE301 - Assignment #1     %%%%%%%%%%%%%%%%%%%%%%

clc; clear all;

[a, b, err_goal, h_n] = deal(pi/6, pi/2, 1e-5, 1);

err = err_goal + 1;

true_int = 1.6724695;

f = @(x)(cos(x)*(exp(x)+x))/(x^2-log(x));

%%%% Compound mid-point:

while err > err_goal

    h = (b-a)/h_n;
    sum = 0;

    for k = 1:h_n

        z_i = a +((k-1)*h)+(h/2);
        nth_section = h*f(z_i);
        sum = sum + nth_section;

    end

    err = abs(sum - true_int);
    h_n = h_n + 1;

end

comp_mp_h = h_n - 1;

%%%% Reset starting conditions

```

```
h_n = 1;
err = err_goal + 1;


%%%% Compound trapezoid:

%       (a+h*(k-1))+(a+h*k)
%  h * -------------------
%                2

while err > err_goal

    h = (b-a)/h_n;
    sum = 0;

    for k = 1:h_n

        x_alpha = a + h*(k-1);
        x_beta = a + h*k;
        nth_section = h * (f(x_alpha)+f(x_beta))/2;
        sum = sum + nth_section;

    end

    err = abs(sum - true_int);
    h_n = h_n + 1;

end

comp_trap_h = h_n - 1;

%%%% Reset starting conditions

h_n = 1;
err = err_goal + 1;


%%%% Simpson's rule:

% (h/3)(A + B + C + D), where A & D are f(x_1), f(x_n)


A = f(a);
D = f(b);


while err > err_goal
```

```
85
86      h = (b-a)/h_n;
87      sum = 0;
88
89      B = 0;
90      C = 0;
91
92      for k = 1:2:(h_n-1)
93
94          B_section = 4*f(a+h*k);
95          B = B + B_section;
96
97      end
98
99
100     for k = 2:2:(h_n-1)
101
102         C_section = 2*f(a+h*k);
103         C = C + C_section;
104
105     end
106
107     sum = (h/3)*(A + B + C + D);
108
109     err = abs(sum - true_int);
110     h_n = h_n + 1;
111
112 end
113
114 simpson_h = h_n - 1;
115
116
117
118 disp("The number of sub-intervals needed for:");
119
120 line_1 = ['----> Compound mid-point: ', num2str(comp_mp_h)];
121 line_2 = ['----> Compound trapezoid: ', num2str(comp_trap_h)];
122 line_3 = ['----> Simpson''s rule:     ', num2str(simpson_h)];
123
124 disp(" ");
125 disp(line_1);
126 disp(line_2);
127 disp(line_3);
```