Madrigal Weersink 3X03 Assignment 2
25/10/2022
400244666

1.

```
function B = inverse(A)
    [m,n] = size(A);
    [L, U, P] = lu(A);
    L_I = eye(n);
    U_I = eye(n);
    U_diag = true;
    L_diag = true;
    for i = 1:n
        U_I(n-i+1,:) = U_I(n-i+1,:)./U(n-i+1,n-i+1);
        U(n-i+1,:) = U(n-i+1,:)./U(n-i+1,n-i+1);
        if i < n
            for j = i+1:n
                U_I(n-j+1,:) = U_I(n-j+1,:) - U(n-j+1,n-i+1).*U_I(n-i+1,:);
                U(n-j+1,:) = U(n-j+1,:) - U(n-j+1,n-i+1).*U(n-i+1,:);
            end
        end
    end
    for i = 1:n
        L_I(i,:) = L_I(i,:)./L(i,i);
        L(i,:) = L(i,:)./L(i,i);
        %disp(L_I)
        if i < n
            for j = i+1:n
                L_I(j,:) = L_I(j,:) - L(j,i).*L_I(i,:);
                L(j,:) = L(j,:) - L(j,i).*L(i,:);
                %disp(L_I)
            end
        end
    end
    B = U_I*L_I*P;
end
```

```
n=  700 time= 1.4e+00 error=3.71e-14
n= 1400 time= 2.7e+01 ratio= 19.4 error=5.38e-14
n= 2800 time= 3.0e+02 ratio= 10.8 error=1.87e-13
n= 5600 time= 1.5e+03 ratio=  5.1 error=9.96e-13
```

2. a)

$$\begin{aligned} det(A) &= 0.1 \cdot \begin{vmatrix} 0.6 & 0.9 \\ 1.5 & 3 \end{vmatrix} - 0.3 \cdot \begin{vmatrix} 0.3 & 0.9 \\ 0.6 & 3 \end{vmatrix} + 0.7 \cdot \begin{vmatrix} 0.3 & 0.6 \\ 0.6 & 1.5 \end{vmatrix} \\ &= 0.1 \cdot 0.45 - 0.3 \cdot 0.36 + 0.7 \cdot 0.09 = 0.045 - 0.108 + 0.063 = 0 \end{aligned}$$

We can calculate the determinant of A to be 0, which tells us that A is singular.

b)

$$A = \begin{bmatrix} 0.1 & 0.3 & 0.7 \\ 0.3 & 0.6 & 0.9 \\ 0.6 & 1.5 & 3 \end{bmatrix} \xrightarrow[R1\ \&\ R3]{swap} \begin{bmatrix} 0.6 & 1.5 & 3 \\ 0.3 & 0.6 & 0.9 \\ 0.1 & 0.3 & 0.7 \end{bmatrix} \xrightarrow[1/2\ R1]{R2-} \begin{bmatrix} 0.6 & 1.5 & 3 \\ 0 & -0.15 & -0.6 \\ 0.1 & 0.3 & 0.7 \end{bmatrix}$$
$$\xrightarrow[1/6\ R1]{R3-} \begin{bmatrix} 0.6 & 1.5 & 3 \\ 0 & -0.15 & -0.6 \\ 0 & 0.05 & 0.2 \end{bmatrix} \xrightarrow{R3+1/3\ R2} \begin{bmatrix} 0.6 & 1.5 & 3 \\ 0 & -0.15 & -0.6 \\ 0 & 0 & 0 \end{bmatrix}$$

Here, we have a row of zeros, and so the process fails.

c) In Matlab, cond(A) returns the number 3.6114e+16 for the matrix A.

When performing Gaussian elimination on the matrix in Matlab, I get this result:

```
0.6000    1.5000    3.0000
0.3000    0.6000    0.9000
0.1000    0.3000    0.7000

0.6000    1.5000    3.0000
     0   -0.1500   -0.6000
0.1000    0.3000    0.7000

0.6000    1.5000    3.0000
     0   -0.1500   -0.6000
0.0000    0.0500    0.2000

0.6000    1.5000    3.0000
     0   -0.1500   -0.6000
0.0000   -0.0000   -0.0000
```

As we can see, the end result includes a couple of -0.0000s. This means that due to rounding errors, this is a very small negative number that is being rounded to zero on my display. In Matlab, the cond function returns the 2-norm condition number for inversion, equal to the ratio of the largest singular value of A to the smallest. Using the svd function, we can see that for this matrix we get $svd(A) = \begin{bmatrix} 3.6646 \\ 0.1760 \\ 0.0000 \end{bmatrix}$. Thus, our ratio should be between 3.6646 and 0.0000. If the zero value were exact, the ratio would be infinite, as any number divided by 0 is infinity, but I've already proven that this isn't exactly 0. As such, cond(A) returns 3.6114e+16 instead of infinity.

d)

Running this code in Matlab:

```
A = [0.1 0.3 0.7; 0.3 0.6 0.9; 0.6 1.5 3];
b = [1.4; 1.8; 6];
x = A\b;
disp(x)
actual_x = [0; 0; 2];
error = norm(x-actual_x,inf)/norm(actual_x,inf);
disp(error)
```

We get the following results:

```
Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND =  1.850372e-18.
> In A2 (line 38)

    0
    0
    2

    0
```

Showing that the result is very accurate, though Matlab does warn that the matrix is close to singular and the results may be innaccurate.

3. a)

$$\Gamma(0.5) = \sqrt{\pi},\ \Gamma(0.75) = \sqrt{\pi/2},\ \Gamma(1) = 1$$
$$p_2(0.5) = c_0 + 0.5c_1 + 0.25c_2 = \sqrt{\pi}$$
$$p_2(0.75) = c_0 + 0.75c_1 + 0.5625c_2 = \sqrt{\pi/2}$$
$$p_2(1) = c_0 + c_1 + c_2 = 1$$
$$\begin{bmatrix} 1 & 0.5 & 0.25 \\ 1 & 0.75 & 0.5625 \\ 1 & 1 & 1 \end{bmatrix}\begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} \sqrt{\pi} \\ \sqrt{\pi/2} \\ 1 \end{bmatrix}$$
$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 1 & 0.5 & 0.25 \\ 1 & 0.75 & 0.5625 \\ 1 & 1 & 1 \end{bmatrix}^{-1}\begin{bmatrix} \sqrt{\pi} \\ \sqrt{\pi/2} \\ 1 \end{bmatrix}$$

Using this Matlab script:

```
X = [1 0.5 0.25; 1 0.75 0.5625; 1 1 1];
Y = [sqrt(pi); sqrt(pi/2); 1];
C = X\Y;
disp(C)
```

We get $\begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 3.6082 \\ -4.7348 \\ 2.1266 \end{bmatrix} \rightarrow \Gamma(x) = 3.6082 - 4.7348x + 2.1266x^2$.

b)

$$\Gamma(x) = 3.6082 - 4.7348x + 2.1266x^2 = 1.5$$
$$2.1266x^2 - 4.7348x + 2.1082 = 0$$
$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} = \frac{-(-4.7348) \pm \sqrt{(-4.7348)^2 - 4(2.1266)(2.1082)}}{2(2.1266)} = 1.6116, 0.6155$$

4. a)

$$\Gamma(-1) = 1,\ \Gamma(0) = 0,\ \Gamma(1) = 1$$
$$p_2(-1) = c_0 - c_1 + c_2 = 1$$

$$p_2(0) = c_0 = 0$$
$$p_2(1) = c_0 + c_1 + c_2 = 1$$
$$\begin{bmatrix} 1 & -1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$
$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

```
X = [1 -1 1; 1 0 0; 1 1 1];
Y = [1; 0; 1];
C = X\Y;
disp(C)
```

$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix},\ p_2 = x^2$$

b)

$$(-1, 1),\ (0, 0),\ (1, 1)$$
$$L_i(x) = \prod_{0 \leqslant m \leqslant k,\ m \neq j\}} \frac{x - x_m}{x_j - x_m}$$
$$L_0(x) = \frac{(x-0)(x-1)}{(-1-0)(-1-1)} = \frac{x^2 - x}{2}$$
$$L_1(x) = \frac{(x+1)(x-1)}{(0+1)(0-1)} = \frac{x^2 - 1}{-1}$$
$$L_2(x) = \frac{(x+1)(x-0)}{(1+1)(1-0)} = \frac{x^2 + x}{2}$$
$$p_2(x) = y_0 L_0(x) + y_1 L_1(x) + y_2 L_2(x) = 1 \cdot \frac{x^2 - x}{2} + 0 \cdot \frac{x^2 - 1}{-1} + 1 \cdot \frac{x^2 + x}{2} = x^2$$
$$p_2 = x^2$$

c)

$$(-1, 1),\ (0, 0),\ (1, 1)$$
$$\phi_j(x) = \prod_{i=0}^{j-1} (x - x_x)$$
$$p_2(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) = f(x_i)$$
$$p_2(x_0) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) = f(x_0)$$
$$c_0 = f(x_0) = 1$$
$$p_2(x_1) = c_0 + c_1(x_1 - x_0) + c_2(x_1 - x_0)(x_1 - x_1) = f(x_1)$$
$$c_0 + c_1(x_1 - x_0) = f(x_1)$$
$$c_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} = \frac{0 - 1}{0 - (-1)} = -1$$

$$p_2(x_2) = c_0 + c_1(x_2 - x_0) + c_2(x_2 - x_0)(x_2 - x_1) = f(x_2)$$

$$c_2 = \frac{\frac{f(x_2)-f(x_1)}{x_2-x_1} - \frac{f(x_1)-f(x_0)}{x_1-x_0}}{x_2 - x_0} = \frac{\frac{1-0}{1-0} - \frac{0-1}{0+1}}{1+1} = 1$$

$$p_2(x) = f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1)$$

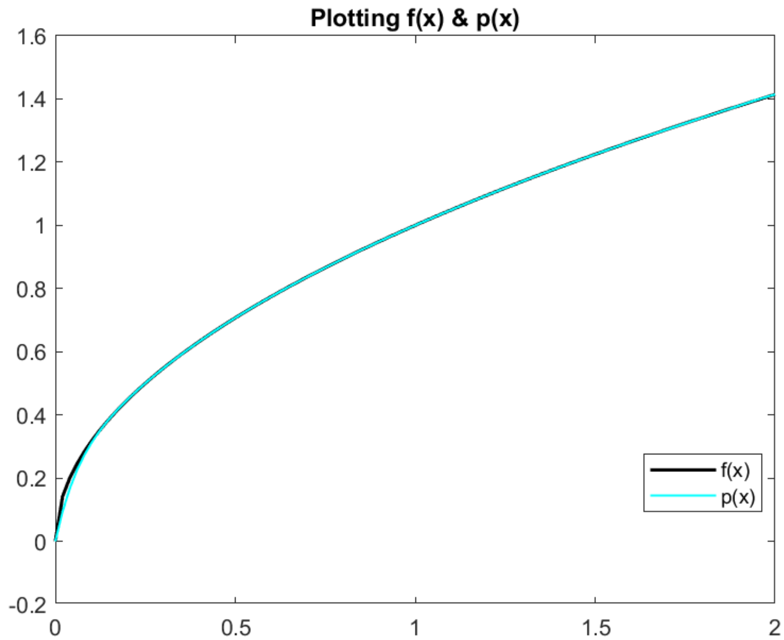
$$= 1 + (-1)(x + 1) + (1)(x + 1)(x - 0) = 1 - x - 1 + x^2 + x = x^2$$

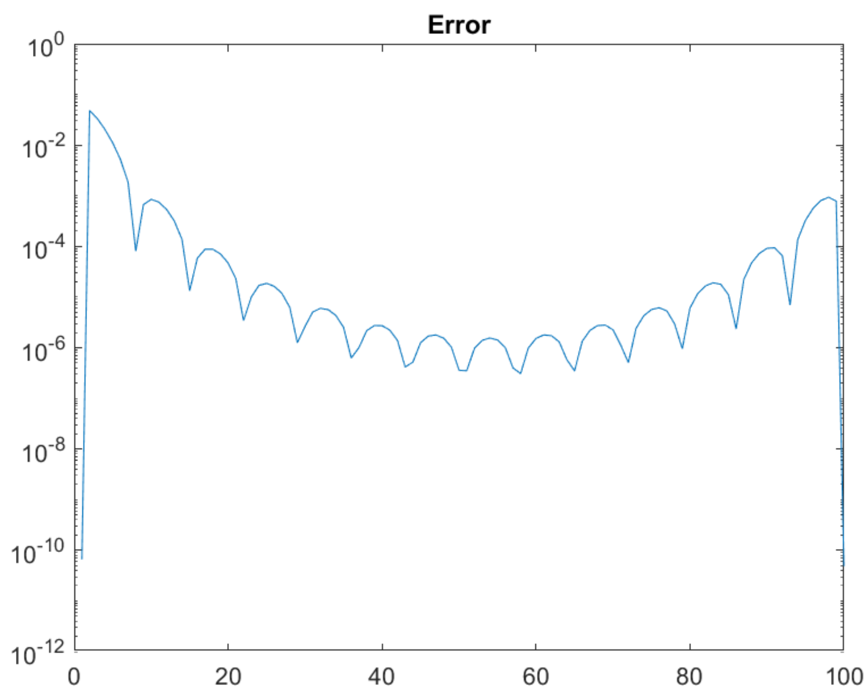
$$p_2 = x^2$$

For each method $p_2 = x^2$.

5. a)

```
f = @(x) (sqrt(x));

x_15 = linspace(0,2,15);
f_15 = f(x_15);
p = polyfit(x_15, f_15, 14);

x_100 = linspace(0,2,100);
f_100 = f(x_100);
p_100 = polyval(p, x_100);

plot(x_100, f_100, "Color", 'k', 'LineWidth', 1.5)
hold on
plot(x_100, p_100, "Color", 'c', 'LineWidth', 1)
legend('f(x)', 'p(x)','Location','best')
title('Plotting f(x) & p(x)')
hold off

figure
semilogy(abs(f_100-p_100))
hold on
title('Error')
hold off
```



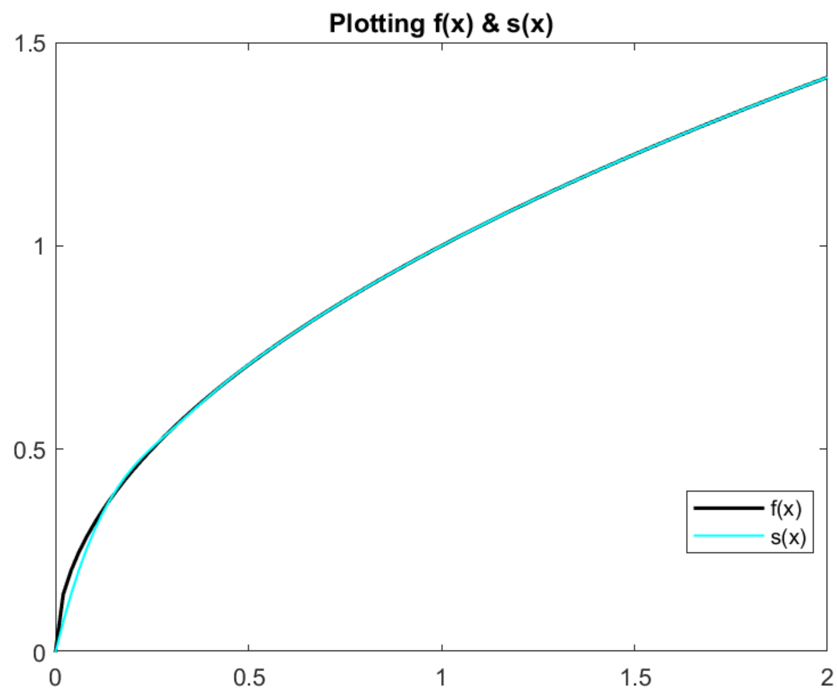


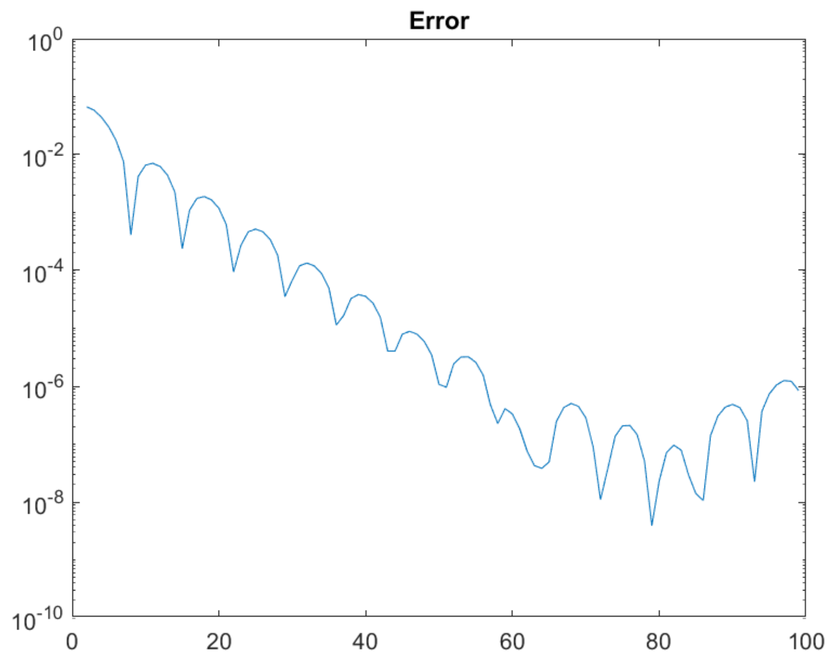
The error is largest when x is close to zero because using interpolation creates a function that doesn't naturally curve the way that the original function does. The error bounces up and down because the low points are the interpolated points on the graph, while each peak is the area in between where the curve of the interpolated function does not match up with the original. This is most extreme near zero, as you can see if you look closely at the graph, and so the error is high.

b)

```
f = @(x) (sqrt(x));
x_15 = linspace(0,2,15);
```

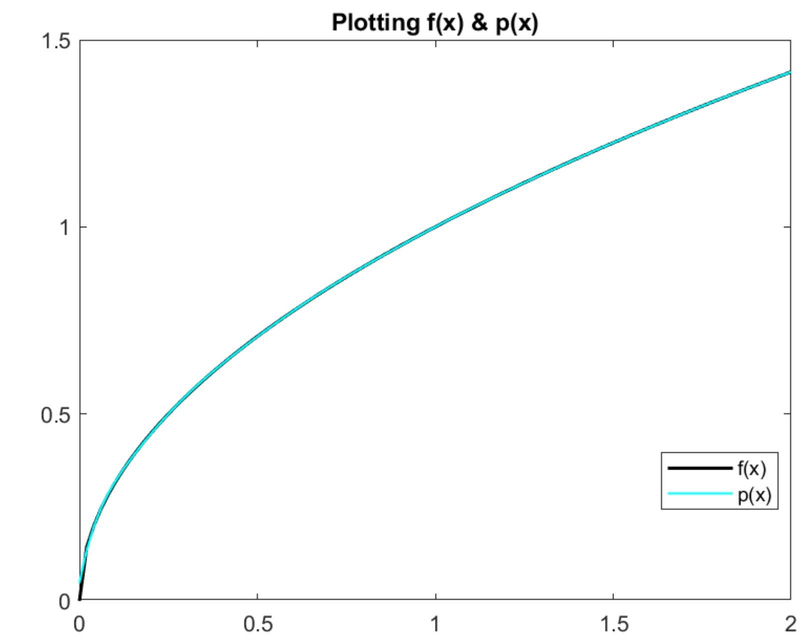
```
f_15 = f(x_15);

x_100 = linspace(0,2,100);
f_100 = f(x_100);
s = spline(x_15, f_15, x_100);

plot(x_100, f_100, "Color", 'k', 'LineWidth', 1.5)
hold on
plot(x_100, s, "Color", 'c', 'LineWidth', 1)
legend('f(x)', 'p(x)','Location','best')
title('Plotting f(x) & p(x)')
hold off

figure
semilogy(abs(f_100-s))
hold on
title('Error')
hold off
```

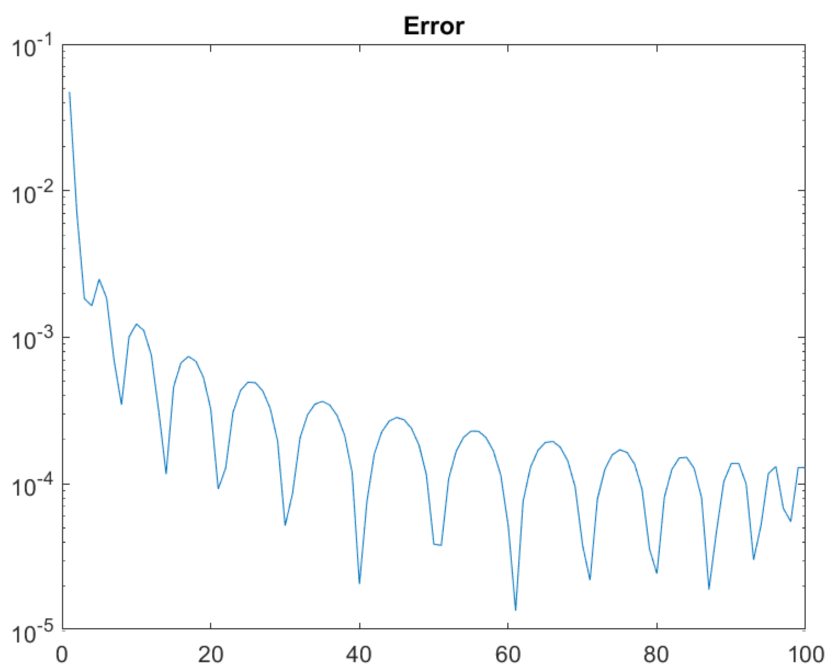




c)

```
f = @(x) (sqrt(x));
i = linspace(0,14,15);
c_x_15 = (0+2)/2 + ((2-0)/2)*cos(((2*i+1)/(2*14+2)*pi));

x_15 = linspace(0,2,15);
f_15 = f(c_x_15);
p = polyfit(c_x_15, f_15, 14);

x_100 = linspace(0,2,100);
f_100 = f(x_100);
p_100 = polyval(p, x_100);

plot(x_100, f_100, "Color", 'k', 'LineWidth', 1.5)
hold on
plot(x_100, p_100, "Color", 'c', 'LineWidth', 1)
legend('f(x)', 'p(x)','Location','best')
title('Plotting f(x) & p(x)')
hold off

figure
semilogy(abs(f_100-p_100))
hold on
title('Error')
hold off
```





d) In both a) and c), the error fluctuates in time with the 15 points the function is interpolated from. However, in a) the general trend of the function follows a somewhat parabolic curve, except at each end, where it sharply drops down. In c), the trend follows more of a gentle curve downwards, evening out at $\sim 10^{-4}$. It is highest near zero, and decreases throughout. The error in c) is overall higher, though, with it's minimum error being $\sim 10^{-5}$, and a)'s minimum error being around $\sim 10^{-10}$, or $\sim 10^{-6}$ if we disregard the edges of the function. They do have similar maximum values, both being $\sim 10^{-1}$.

6.

$$h = \frac{b-a}{n} = \frac{1-0}{2} = \frac{1}{2}$$

$$M = \max_{0 \leqslant x \leqslant 1} |f'''(x) - p_2(x)| = \max|\left(8x^3 + 12x\right)e^{x^2} - p_2(x)|$$

$$f'''(x) = \left(8x^3 + 12x\right)e^{x^2} \leqslant 20e \; on \; [0, 1]$$

$$|e^{x^2} - p_2(x)| \leqslant \frac{M}{4(n+1)}h^{n+1}$$

$$|e^{x^2} - p_2(x)| \leqslant \frac{20e}{4(2+1)}\left(\frac{1}{2}\right)^3 \approx 0.5663087$$