**Michael Goforth**
**CAAM 550**
**HW 8**
**10/27/2021**

**Problem 1**

**part a**
See Jupyter notebook for code and results.

**part b**
See Jupyter notebook for code and results.

**part c**
See Jupyter notebook for code and results.

**part d**
See Jupyter notebook for code and results.

**part e**
See Jupyter notebook for code and results.

**Problem 2**

**part a**
See Jupyter notebook for code and results.

**part b**
See Jupyter notebook for code and results. The root approximation is about 1.875.

**Problem 3**

**part a**

$$\sum_{k=1}^{n}\sum_{l=1}^{m} a_{kl} p_k(x_i) q_l(y_j) = f(x_1, y_j),\ i = 1, \ldots, n,\ j = 1, \ldots, m$$

$$p_i(x) = L_i(x) = \prod_{\substack{j=1 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j},\ i = 1, \ldots, n$$

$$q_i(y) = L_i(y) = \prod_{\substack{j=1 \\ j\neq i}}^{m} \frac{y - y_j}{y_i - y_j},\ i = 1, \ldots, m$$

For any $(x_i, y_j) \in [a, b] \times [c, d],\ i = 1, \ldots, n,\ j = 1, \ldots, m,$

$$p_k(x_i) = \begin{cases} 1, & \text{if } k = i \\ 0, & \text{if } k \neq i \end{cases}$$

$$q_l(y_j) = \begin{cases} 1, & \text{if } l = j \\ 0, & \text{if } l \neq j \end{cases}$$

$$f(x_1, y_j) = \sum_{k=1}^{n}\sum_{l=1}^{m} a_{kl} p_k(x_i) q_l(y_j) = a_{kl}$$

See Jupyter notebook for code and further results.

**part b**

$$\sum_{k=1}^{n}\sum_{l=1}^{m} a_{kl} p_k(x_i) q_l(y_j) = f(x_1, y_j),\ i = 1, \ldots, n,\ j = 1, \ldots, m$$

$$p_i(x) = N_i(x) = \prod_{j=1}^{i-1} (x - x_j),\ i = 1, \ldots, n$$

$$q_i(y) = N_i(y) = \prod_{j=1}^{i-1} (y - y_j),\ i = 1, \ldots, m$$

$$N_1(x) = N_1(y) = 1$$

$$p_i(x_k) = \prod_{j=1}^{i-1} (x_k - x_j) = 0 \text{ if } k > i$$

$$q_i(y_l) = \prod_{j=1}^{i-1} (y_l - y_j) = 0 \text{ if } l > i$$

$$a_{kl} = \frac{f(x_k, y_l) - \sum_{i=1}^{k-1} \sum_{j=1}^{l-1} a_{ij} p_i(x_k) q_j(y_l)}{p_k q_l}$$

By starting with a matrix $A \in \mathbb{R}^{n\times m}$ of all zeroes, the equations above can be used to find elements of $A$ one by one starting at the top left and working down the matrix one row at a time.

**Problem 4**
**part a**
$\prod_n f = p_n$ where $p_n$ is a polynomial of degree $n$. Then since the polynomial of degree $n-1$ going through $n$ different points is unique, $\prod_n p_n = p_n$. Therefore $(\prod_n)^2 f = p_n$ and therefore $\prod_n = (\prod_n)^2$ and $\prod_n$ is a projector.

**part b**
$\prod_0 f = p_0 = f(x_1)$ for any function $f$ and given point $x_1$. Therefore the $||\prod_0 f||_{L^\infty([a,b])} = f(x_1)$. Then

$$\max_{0\neq f\in\mathbb{C}([a,b])} \frac{||\Pi_0 f||_{L^\infty([a,b])}}{||f||_{L^\infty([a,b])}} = 1$$

since $||f||_{L^\infty([a,b])} \geq |f(x_1)|$.
Similarly, because a polynomial of degree 1 is a line, if $x_0 = a$ and $x_1 = b$, then $||\Pi_1 f||_{L^\infty([a,b])} = max|f(a)|, |f(b)|$. Again since $||f||_{L^\infty([a,b])} \geq max|f(a)|, |f(b)|$,

$$||\Pi_1||_{L^\infty([a,b])} = \max_{0\neq f\in\mathbb{C}([a,b])} \frac{||\Pi_1 f||_{L^\infty([a,b])}}{||f||_{L^\infty([a,b])}} = 1$$

**part c**

$$||Pi_n||_{L^\infty([a,b])} = \max_{||f||_{L^\infty([a,b])}=1} ||Pi_n f||_{L^\infty([a,b])}$$

$$||Pi_n||_{L^\infty([a,b])} = \max_{||f||_{L^\infty([a,b])}=1} ||\sum_{j=0}^{n} f(x_j)L_j(x)||_{L^\infty([a,b])}$$

$$||Pi_n||_{L^\infty([a,b])} = \max_{||f||_{L^\infty([a,b])}=1} \left[\max_{x\in[a,b]} |\sum_{j=0}^{n} f(x_j)L_j(x)|\right]$$

Because $||f||_{L^\infty([a,b])} = 1$, a function can be constructed with $f(x_j) = \pm 1$ and $max_{x\in[a,b]}|f(x)| = 1$ (chosen so that the sign of all the terms is the same) which maximizes the sum in the brackets. Therefore

$$||Pi_n||_{L^\infty([a,b])} = \max_{x\in[a,b]} \sum_{j=0}^{n} |L_j(x)|$$

**part d**
Using the triangle inequality

$$||f - p_n||_{L^\infty([a,b])} \leq ||f - p_*||_{L^\infty([a,b])} + ||p_* - p_n||_{L^\infty([a,b])}$$

Because $\Pi_n$ is a projector (shown in part a),

$$\begin{aligned}
||f - p_n||_{L^\infty([a,b])} &\leq ||f - p_*||_{L^\infty([a,b])} + ||\Pi_n(p_* - f)||_{L^\infty([a,b])} \\
||f - p_n||_{L^\infty([a,b])} &\leq ||f - p_*||_{L^\infty([a,b])} + ||\Pi_n||_{L^\infty([a,b])}||(p_* - f)||_{L^\infty([a,b])} \\
||f - p_n||_{L^\infty([a,b])} &\leq (1 + ||\Pi_n||_{L^\infty([a,b])})||f - p_*||_{L^\infty([a,b])}
\end{aligned}$$

**part e**
Consider the polynomial $p_*$ that minimizes $||f - p_n||_{L^\infty([a,b])}$, then from part d above

$$||f - p_n||_{L^\infty([a,b])} \leq (1 + ||\Pi_n||_{L^\infty([a,b])})||f - p_*||_{L^\infty([a,b])}$$

The Lebesgue Constant is acting as a measure of the error of the calculated interpolant and the best possible interpolant, the smaller the constant, the closer the calculated interpolation is to the best interpolation.

**Problem 5**

**part a**
See Jupyter notebook for code and results.

**part b**
See Jupyter notebook for code and results.

**part c**
See Jupyter notebook for code and results.

**part d**
See Jupyter notebook for code and results. The error bound for linear spline interpolation holds for the bounded twice differentiable functions ($\sin(x)$, $\sin(30x)$, and $e^x$) as written. However the second derivative of $x^{4/3}$, and $\sqrt[3]{x - 1/2}$ are unbounded so the error bound does not apply to these functions. Finally the second derivative of $|x - 0.567|$ is undefined at $x = 0.567$ so the error for this function also does not follow the error bound.
The error bound of $\sin(3x)$ and $\sin(30x)$ both converge $O(h^2)$, but $sin(30x)$ converges slower due to the larger maximum value of its second derivative on the interval [0,1].

**part e**

See Jupyter notebook for code and results. Order of error for the functions:

$$\begin{aligned} f(x) = \sin(3x) &\Rightarrow O(4) \\ f(x) = \sin(30x) &\Rightarrow O(4) \\ f(x) = e^x &\Rightarrow O(4) \\ f(x) = x^{4/3} &\Rightarrow O(2.66) \\ f(x) = |x - 0.567| &\Rightarrow O(2) \\ f(x) = \sqrt[3]{x - 1/2} &\Rightarrow O(2/3) \end{aligned}$$

I determined these values by making plots of $L_i n f$ error vs $n$ and $h^p$ vs $n$ for each function and incrementing $n$ up and down until the plot approached linear.