1. Consider the points

| $x$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| $y$ | 4.0 | 3.1 | 2.8 | 2.5 | 2.3 | 2.2 | 2.1 | 2.0 |

   (a) Use the Discrete Cosine Transform to find a wave function that interpolates all eight points.

   (b) Find another wave function that approximates the points by eliminating the two highest frequencies in your previous answer. Plot both wave functions on the same graph along with the original points.

2. (*Compressing a Grayscale Image*) Download the file 'lion.jpg', and import it into Julia using 'img=load("lion.jpg")'. Convert the image into a grayscale image with 'img2=Gray.(img)'. Then convert the image into a matrix A.

   (a) After level shifting the matrix, and applying the DCT to each 8x8 block, apply a low pass filter by letting $y_{ij} = 0$ whenever $i + j \geq 4$. Use the Inverse DCT and then undo the level shifting to construct an approximation matrix $\bar{A}$. Compare the compressed image with the original image.

   (b) Construct another approximation matrix $\bar{\bar{A}}$ by using the singular value decomposition to compute a rank 30 approximation of $A$. Repeat with a rank 20 approximation. Compare both compressed images with the original image.

3. (*Compressing a Color Image*) Download the file 'colors.jpg', and import it into Julia. Decompose the image matrix $A$ into its red, green, and blue component matrices. Make a change of variables to $Y, U, V$ components given by

$$Y = .299R + .587G + .114B, \quad U = B - Y, \quad V = R - Y.$$

   Apply luminance quantization to $Y$ (p.255) and chrominance quantization to $U$ and $V$ (p.258) to get approximations $\bar{Y}, \bar{U}$, and $\bar{V}$. Change variables back to red, green, and blue components $\bar{R}, \bar{G}$, and $\bar{B}$, and then reconstruct the compressed image $\bar{A}$ from its components. Compare the compressed image $\bar{A}$ with both the original $A$ and the difference $A - \bar{A}$.

4. (*Huffman Coding*) Consider the following frequency distribution for the symbols $A, E, H, L, S, \_$ where $\_$ denotes a space.

| A | E | H | L | S | _ |
|---|---|---|---|---|---|
| 1 | 4 | 2 | 4 | 6 | 2 |

   (a) Use Huffman Coding to construct a variable length code for these symbols corresponding to these frequencies.

   (b) Show the tree that you would use to build the code. (*You can draw the tree by hand and take a picture or scan it. I haven't yet figured out an easy way to make Julia draw the tree.*)

   (c) Find the code that would correspond to the text "SHE SELLS SEASHELLS".

   (d) What percentage of storage space is saved in using your variable length code to store "SHE SELLS SEASHELLS" instead of a fixed length code?