**Problem 1** (10 points) Consider an alphabet with two symbols \( \{ A, B \} \), where \( P(A) = x \) and \( P(B) = 1 - x \). Plot the entropy as a function of \( x \).

**Problem 2** (15 points) Consider an alphabet with three symbols \( \{ A, B, C \} \), where \( P(A) = x \), \( P(B) = y \) and \( P(C) = 1 - x - y \). Plot the entropy as a function of \( x \) and \( y \).

**Problem 3** (10 points) From the results of Problem 1 and Problem 2, we can expect that the entropy of a source reaches its maximum when all symbols are equally probable. Please prove this formally for a alphabet with \( N \) symbols.

**Problem 4** (15 points) Based on the RGB-to-YUV conversion in Lecture Notes on H.261, derive the YUV-to-RGB conversion in the matrix/vector form.

**Problem 5** (20 points) This is an exercise of arithmetic coding. Given the following probabilities of symbols \( A \), \( B \), \( C \), and \( D \) at different time instants:

<table>
<thead>
<tr>
<th></th>
<th>( t = 1 )</th>
<th>( t = 2 )</th>
<th>( t = 3 )</th>
<th>( t = 4 )</th>
<th>( t = 5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A )</td>
<td>0.5</td>
<td>0.4</td>
<td>0.5</td>
<td>0.4</td>
<td>0.3</td>
</tr>
<tr>
<td>( B )</td>
<td>0.3</td>
<td>0.2</td>
<td>0.2</td>
<td>0.4</td>
<td>0.3</td>
</tr>
<tr>
<td>( C )</td>
<td>0.1</td>
<td>0.2</td>
<td>0.2</td>
<td>0.1</td>
<td>0.2</td>
</tr>
<tr>
<td>( D )</td>
<td>0.1</td>
<td>0.2</td>
<td>0.1</td>
<td>0.1</td>
<td>0.2</td>
</tr>
</tbody>
</table>

we want to encode the sequence "BCAAD" using Implementation #1 as described in class. Consider "D" as the EOF symbol. You don’t need to write a program to do this. Simply compute the interval \([\text{low}, \text{high})\) after each symbol is processed, and summarize the result in a table. Note that all \( \text{cum}_\text{freq}[i] \) are time-varying. At the end of the five symbols, pick a value in the range \([\text{low}, \text{high})\) and send it to the decoder. Then, based on value, the decoder can recover the symbols. Compute the interval \([\text{low}, \text{high})\) after each symbol is processed and summarize the result in a table. Verify that the interval \([\text{low}, \text{high})\) at the encoder varies in synchronization with the interval \([\text{low}, \text{high})\) at the decoder.

**Problem 6** (30 points) Repeat Problem 5 with Implementation #2. Use the following numbers:

\[
\begin{align*}
C &= 8 \\
\text{cum}_\text{freq}[0] &= 10
\end{align*}
\]

Again, at the encoder, compute the interval \([\text{low}, \text{high})\) and the bits to output, after each symbol is processed. At the decoder, compute value, cum, and \([\text{low}, \text{high})\) after each symbol is decoded.