1. **HMM Inference.** The following figure specifies a HMM model. There are 3 possible states, *Onset*, *Mid*, and *End*. Note that the state for the very first observation is always *Onset*, i.e., \( P(S_1 = \text{Onset}) = 1 \). There is also an extra *Final* state. If the length of an observation sequence is \( T \), then the state at time step \( T+1 \) is always *Final*. Please calculate the most probable state sequence for the observation sequence \([C_1, C_2, C_3, C_4, C_5, C_6, C_7]\) and its probability. You should draw the “trellis” showing the probability paths through the HMM as shown in class.

   **Phone HMM for [m]:**

   ![Trellis diagram](image)

   **Output probabilities for the phone HMM:**

   
   
<table>
<thead>
<tr>
<th>State</th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
<th>C6</th>
<th>C7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Onset</td>
<td>0.5</td>
<td>0.2</td>
<td>0.3</td>
<td>0.2</td>
<td>0.7</td>
<td>0.5</td>
<td>0.1</td>
</tr>
<tr>
<td>Mid</td>
<td>0.7</td>
<td>0.9</td>
<td>0.1</td>
<td>0.4</td>
<td>0.6</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>End</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>Final</td>
<td>0.7</td>
<td>0.9</td>
<td>0.4</td>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
</tr>
</tbody>
</table>

2. **HAC.** Create by hand the clustering dendrogram for the following samples of ten points in one dimension.

   \( \text{Sample} = (-2.2, -2.0, -0.3, 0.1, 0.2, 0.4, 1.6, 1.7, 1.9, 2.0) \)

   a. Using single link.
   b. Using complete link

   Note that in class we described HAC assuming similarity functions. You should be able to easily revise the definitions to use distance functions instead. And here the Euclidean distance function is used.

3. **Kmeans.** The Kmeans algorithm can be viewed as searching for the clusters \( C_1, C_2, \ldots, C_k \) with mean \( \mu_1, \mu_2, \ldots, \mu_k \) to minimize the following objective function, typically called the sum-of-square-error criterion \( J_e \):

   \[
   J_e = \sum_{i=1}^{k} \sum_{x \in C_i} ||x - \mu_i||^2
   \]

   Prove that in each iteration of the algorithm (before convergence), the objective function \( J_e \) never increases. In particular, prove that in each individual step:
– “assign each point to clusters according to its closest center” and
– “recompute the center of each cluster”

the objective function never increases.

4. Picking $k$ for Kmeans. One shortcoming of Kmeans is that one has to specify the value of $k$. Consider the following strategy for pick $k$ automatically: try all possible values of $k$ and choose $k$ that minimizes $J_e$. Argue why this strategy is a good/bad idea.

5. Gaussian Mixture Models. Let $f(x)$ be a univariate mixture density such that $f(x|\theta_1)$ is a Normal distribution with mean $\mu_1 = 0$ and $\sigma^2 = 1$, and $f(x|\theta_2)$ is a Normal distribution with mean $\mu_2 = 0$ and $\sigma^2 = 0.5$. Thus we can write $f(x)$ as:

$$f(x) = \alpha \frac{e^{-\frac{1}{2}x^2}}{\sqrt{2\pi}} + \frac{1-\alpha}{\sqrt{\pi}} e^{-x^2}$$

where $\alpha$ is the unknown parameter specifying the prior probability of $\theta_1$. Consider that a single sample $x_1$ has been observed. Determine the maximum likelihood estimate of $\alpha$. 